

COMPUTER® junkyard

The best computer game not played on the computer.™



USER MANUAL

You found a box of old software games at a yard sale and can't wait to play them with your friends. To your dismay, modern computers won't run them. So, you all race to build your own vintage computers out of spare parts.

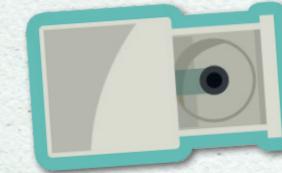
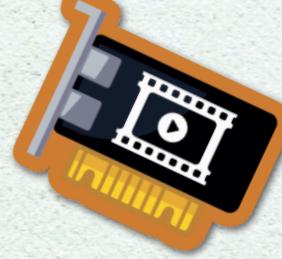
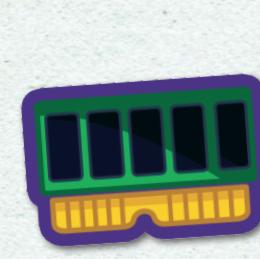
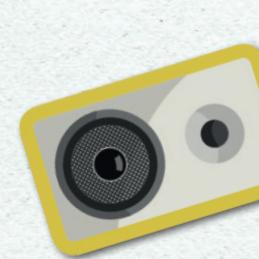
Welcome to *Computer Junkyard*, the perfect place to find all the components you will need!

Scavenge for parts, sabotage, steal, and trade with your opponents, all with just the cash in your pockets. Be the first to build a computer good enough to run your game!

	60-90 Minutes
	2-4 Players
	Ages 14+

No prior computer knowledge required

TABLE OF CONTENTS

	<i>Components</i>	3
	<i>Setup</i>	4
	<i>Objective</i>	6
	<i>How to Play</i>	6
	<i>Phase I: Pre-Build</i>	7
	<i>Phase II: Build the Computer</i>	10
	<i>Phase III: System Check</i>	14
	<i>Winning the Game</i>	14
	<i>Correcting Build Violations</i>	15
	<i>The Finished Computer</i>	15
	<i>Gameplay Variations</i>	16
	<i>Single-Player Mode</i>	16
	<i>Anatomy of a Computer</i>	17
	<i>Frequently Asked Questions</i>	20
	<i>Credits & Thanks</i>	20
	<i>More Information</i>	20

COMPONENTS



4 chassis boards



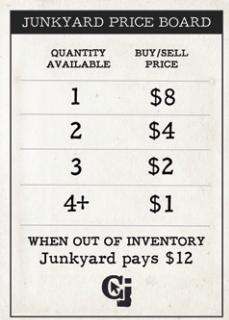
10-sided die



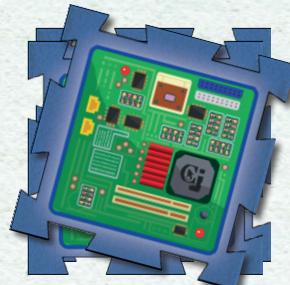
2-minute timer



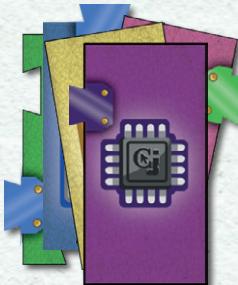
tile pouch



price board
(with stands)



4 motherboards



97 hardware tiles



8 computer bugs



7 software tiles



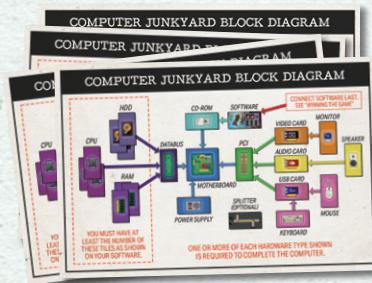
JunkBucks coins



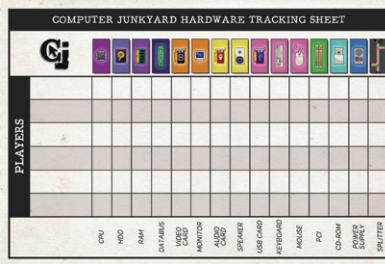
8 lock tokens



8 bug spray tokens



4 block diagrams



pad of player
tracking sheets

SETUP

1. PLAYER ITEMS

Each player takes a **(A) chassis**, **(B) block diagram**, and **(C) \$50 in JunkBucks**.

Select a **banker** to manage the remaining **(D) JunkBucks**.

Players may optionally take a **(E) tracking sheet**.



2. COMMON ITEMS

(F) Place the **die**, **timer**, and **price board** within reach of the players, as well as **2 locks per player** and **2 bug sprays per player**. Discard from the game any remaining locks and sprays.

3. HARDWARE & BUG TILES

Collect all **97 hardware** and **8 bug tiles**. For **4 players**, use all **105 tiles**.

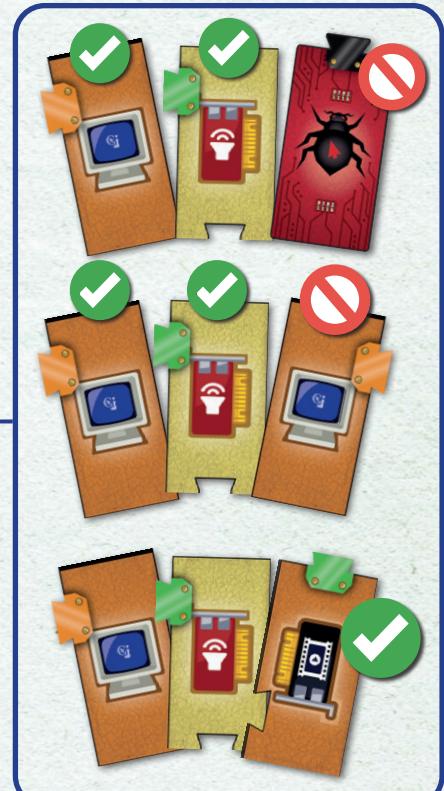
For **3 players**, discard the **21 tiles** with **gold backs**.

For **2 players**, discard the **21 silver-backed** and **21 gold-backed tiles**.

Place the remaining tiles in the **(G) pouch** and mix thoroughly.



Draw **(H) 2 tiles per player** from the **pouch**, placing them face-up into the center of the play area, called the **Junkyard**. Duplicate tiles *can* be in the Junkyard, and should be grouped side-by-side. If any **bugs** are drawn, mix them back into the **pouch** and draw again.



4. PLAYER STARTING TILES

Each player draws **(I) 3 tiles** from the **pouch**, placing them face-up in the area next to their **chassis**, called the **workbench**. If a player draws any **bugs**, or if there are any **duplicate tiles** (tiles of the same type) among the 3 drawn, return the bugs or duplicates to the **pouch** and draw again.

Each player also takes a **(J) motherboard** and places it anywhere on their **chassis**, aligned to the grid. A player's **computer** consists of all tiles **connected back** to the **motherboard** on their **chassis**. Tiles **separated** from the **motherboard** are called **loose**. See the example on the opposite page, and Pages 11-13 for details.

5. SOFTWARE TILES

Gather all **7 software tiles**, and select a set of them as follows:

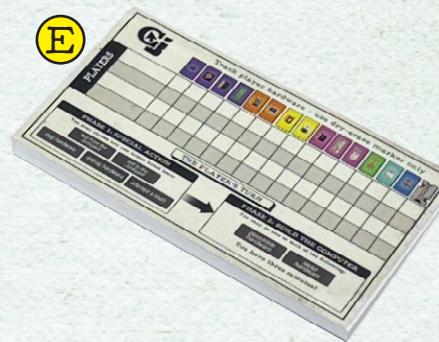
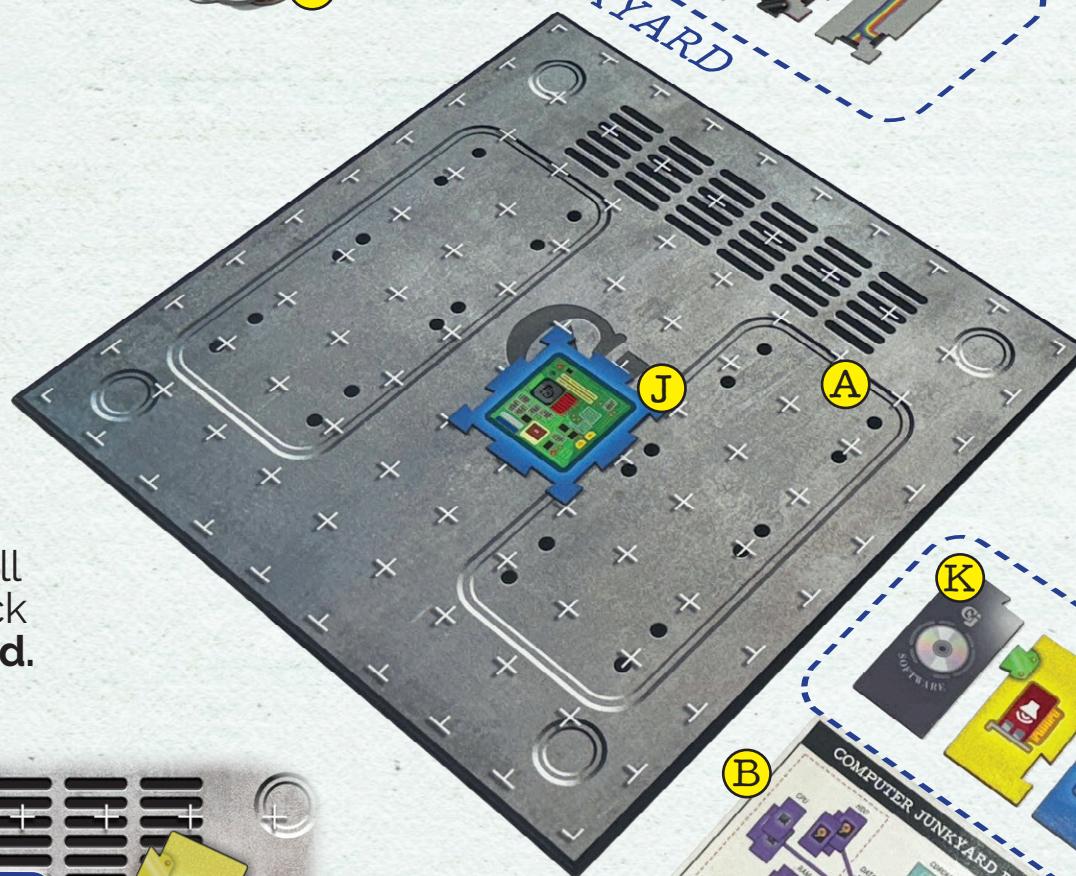
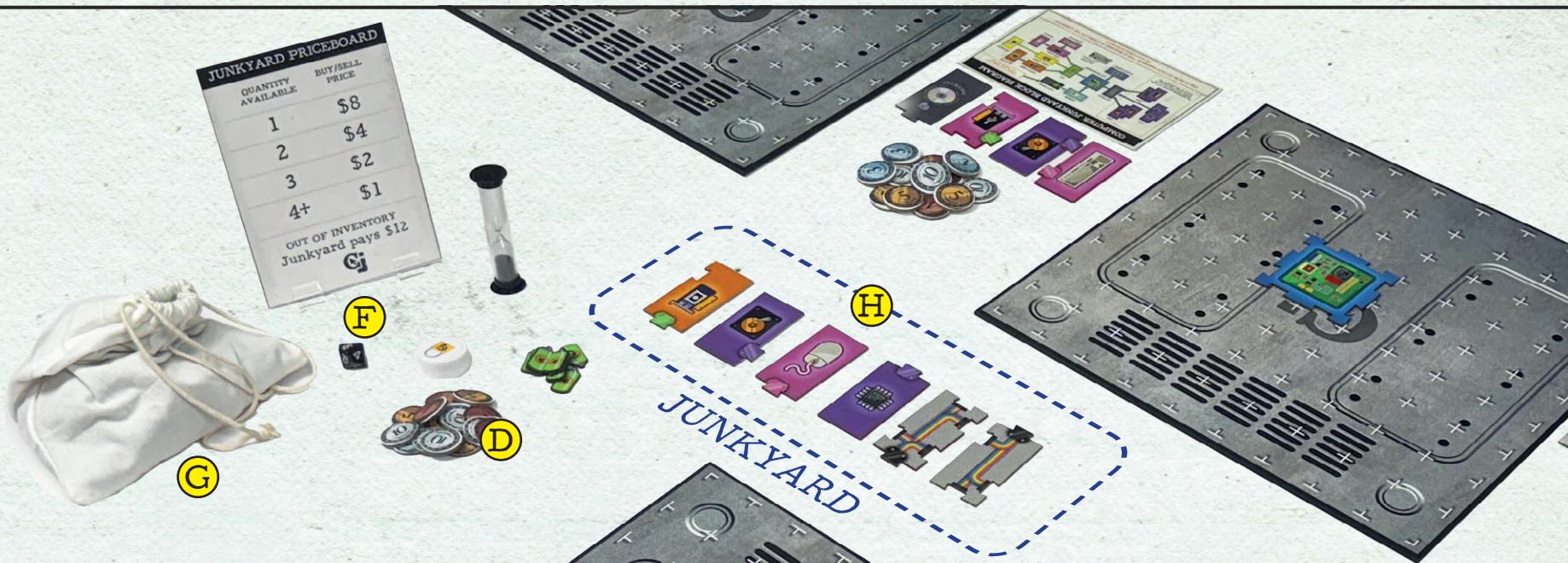
For **4 players**, use all **3** tiles from Group **(A)** & any **1** from Group **(B)**.

For **3 players**, use all **3** tiles from Group **(A)**.

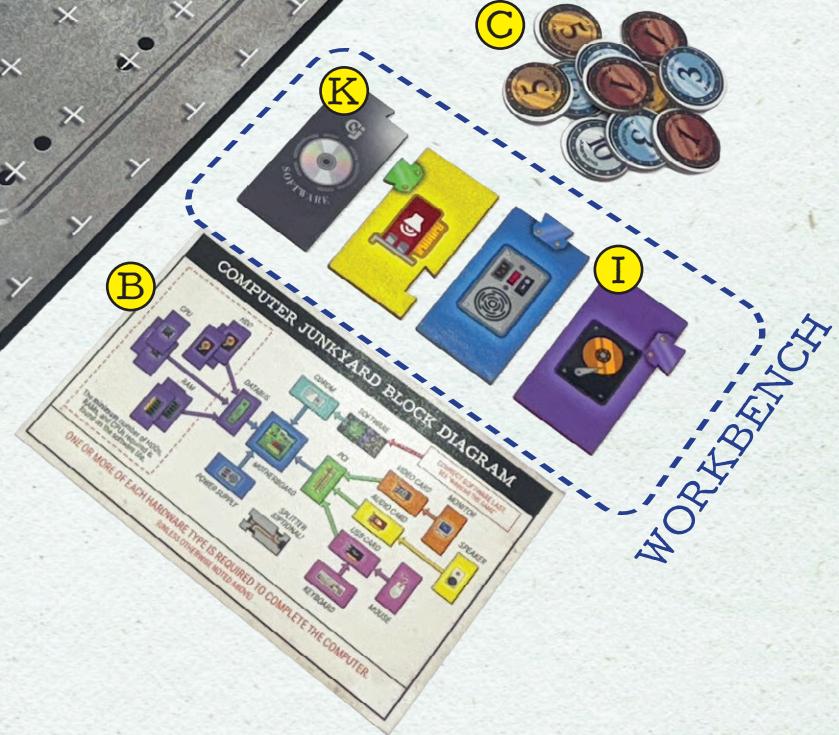
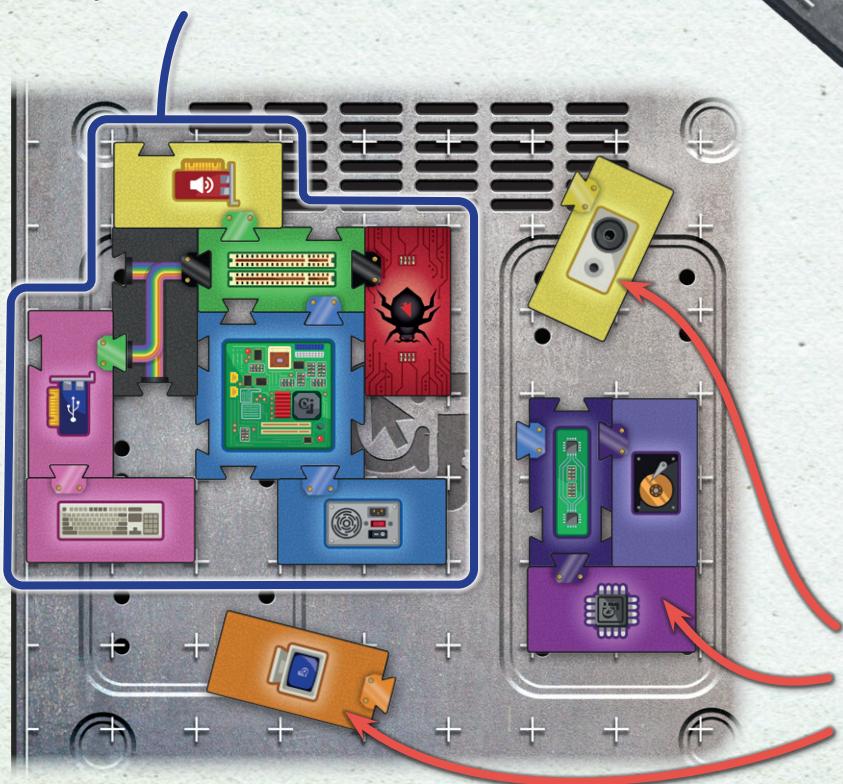
For **2 players**, use any **2** tiles from Group **(B)**.



(K) Take the selected **software tiles**, and give one **face-down** to each player. **Discard** all remaining tiles. Players may view their tile at any time, and may keep it secret from their opponents.



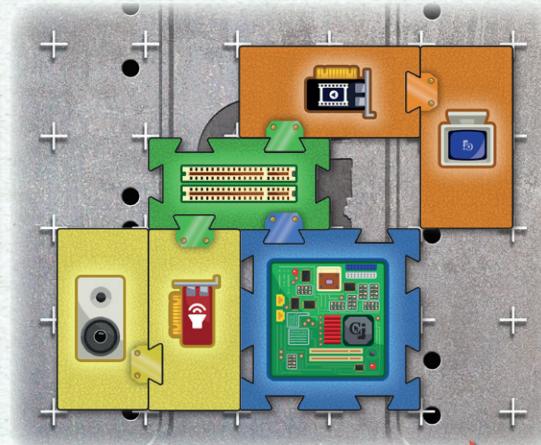
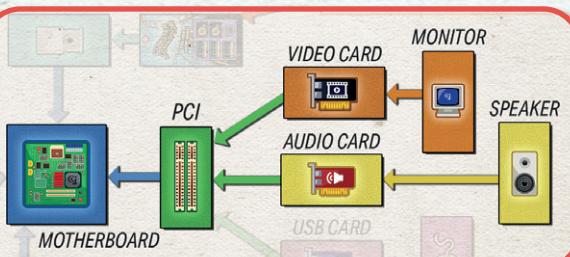
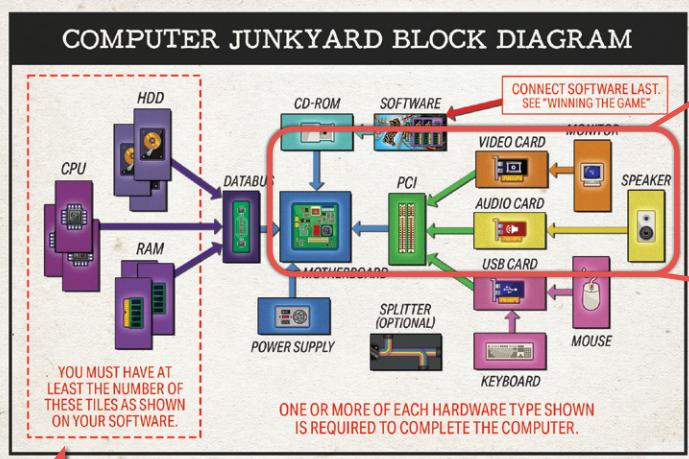
Your **computer** is all tiles connected back to your **motherboard**.



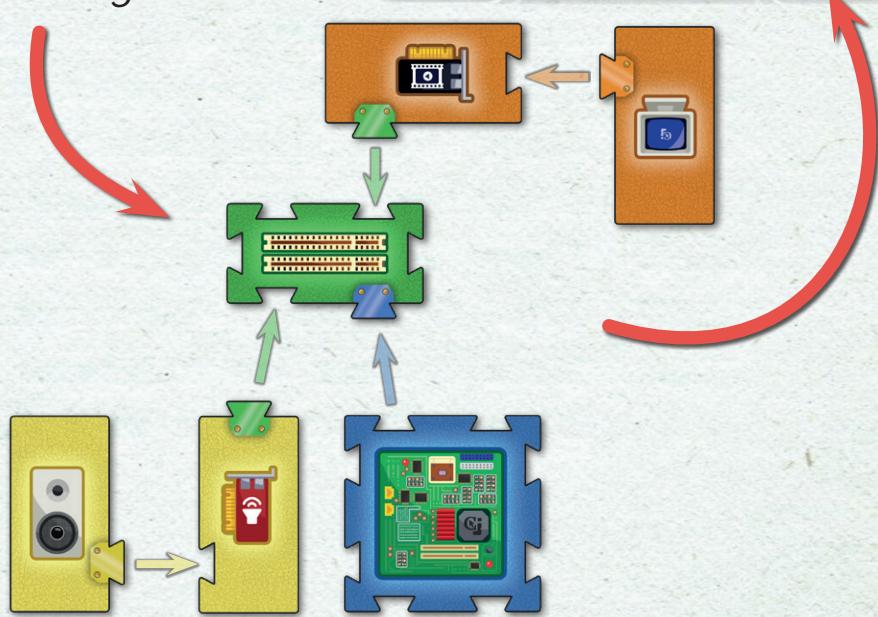
Loose tiles are any tiles that do not connect back to the **motherboard**. While not building your **computer**, loose tiles get moved to your **workbench** (see **Page 13**).

OBJECTIVE

Players compete to build a **computer** by connecting **hardware tiles** together according to the **block diagram**. The winner is the first player to connect their **software tile** to their **computer** which must be **bug-free**, include one of every type of **hardware**, and meet the software's **requirements**.



Here is just one example of how you could build this portion of the block diagram:



HOW TO PLAY

SELECT A SYSTEM ADMINISTRATOR:

Select a player to be the first System Administrator, or **Sysadmin**, and give them the **timer**. The **Sysadmin** will rotate each round.

THE GAME ROUNDS

Computer Junkyard is played in rounds, with each round consisting of 3 phases (see next page). Rounds continue until a player has won the game during a Phase III.

HOW TO PLAY (cont'd.)

PHASE I *Pre-Build*

Each player takes a turn to draw tiles, take one of 3 possible actions, make deals with their opponents, and prepare to build their computer.

PHASE II *Build the Computer*

All players build their computers at the same time, racing against each other and the clock, until time runs out.

PHASE III *System Check*

If a player installed their software during Phase II in an attempt to win, players check whether their computer is built correctly. Otherwise, return to Phase I for a new round.

PHASE I: PRE-BUILD

Starting with the **Sysadmin**, and **continuing clockwise**, each player takes one turn. A turn consists of **drawing tiles** (see below), followed by **taking an action** (Page 8). You may also **make deals** with other players at any time (Page 10).



DRAW TILES

To begin your turn, draw **2 tiles** (or as many that remain) from the **pouch** and place them face-up in the central **Junkyard** (shown on Page 5).

If you draw a **bug** tile, you must play it immediately (see below). If both tiles you draw are **bugs**, return the second to the **pouch** and draw again. If there are only bugs remaining in the pouch, do not draw a second tile (the next player will draw one on their turn).



PLAYING A BUG

To play a **bug**, connect it to an available **port** of your choice on any opponent's **computer** (see **Connecting Tiles**, Page 11). You **may not** rearrange or slide a **computer** to connect the bug. If the bug cannot be connected anywhere, **discard** it from the game. You **may** connect the bug to your own **computer**, but are **not** required to if you cannot connect it anywhere else. Playing a bug creates **bugged hardware**; see Page 12 for details.

PHASE I: PRE-BUILD (cont'd)

TAKE AN ACTION

After you have drawn 2 tiles, you may take any **one** of the following actions:

OPTION
1

Visit the Junkyard

Buy and sell hardware to build your computer.

OPTION
2

Debug the Computer

Try to get rid of bugs that have been connected to your computer.

OPTION
3

Attempt a Theft

Try to steal a tile from an opponent's computer or workbench.

OPTION
1

Visit the Junkyard

You may do any/all of the following:

- **Buy** up to **2 hardware tiles** from the **Junkyard**, and/or
- **Sell** up to **2 hardware tiles** to the **Junkyard**, and/or
- **Sell** one or more **locks** from your **computer** to the **Junkyard** for **\$5** each. **Locks cannot be purchased**; they can only be acquired after a theft (see next page).

BUYING HARDWARE

The price of a tile is based on how many of that **tile type** are available in the **Junkyard**. Pay the banker the amount shown on the **price board**, and place the purchased tile in your **workbench**.

Example: There are 3 Keyboard tiles currently in the Junkyard. You select one of the Keyboard tiles, pay \$2 for it, and place it in your **workbench**. The price for another Keyboard is now \$4.

QUANTITY AVAILABLE	BUY/SELL PRICE
1	\$8
2	\$4
3	\$2
4+	\$1

WHEN OUT OF INVENTORY
Junkyard pays \$12
GJ

SELLING HARDWARE

You may sell tiles from your **computer** or **workbench**. The banker will give you the amount shown on the **price board**, based on how many of the tile type are **currently** in the **Junkyard**, **not** including the tile you are selling. If **none** of the tile type are currently in the **Junkyard**, you receive **\$12** for the sale. Place the sold tile face-up in the Junkyard.

If selling a tile from your **computer** causes other tiles to become **loose** (separated from the **motherboard**; see **Page 5**), move any loose tiles to your **workbench**.

You cannot sell bugs or bugged hardware from your computer (see Page 12).

Example: There is 1 RAM tile in the Junkyard. You sell a RAM from your **workbench**. You receive \$8 and place the tile in the Junkyard. The sale price for another RAM is now \$4.

QUANTITY AVAILABLE	BUY/SELL PRICE
1	\$8
2	\$4
3	\$2
4+	\$1

WHEN OUT OF INVENTORY
Junkyard pays \$12
GJ

INVENTORY IN THE GAME

The total number of each tile type can be found on the **back of the price board**, and is based upon the number of players (see **Setup, Page 4**). You can use this information to determine what is still in the **pouch** during gameplay.

PHASE I: PRE-BUILD (cont'd)

OPTION
2

Debug the Computer

Try to remove **bugs** from your **computer**. You can't win the game with a buggy computer! (See **Page 14** for win conditions)

Repeat the following once for each bug on your own computer:



Roll the die. If the roll is a **10**, remove the **bug** and immediately connect it to an opponent's **computer** (See *Playing a Bug, Page 7*). If you did *not* roll a **10**, **add** to your roll the number of **bug sprays** on your **motherboard** (if any). If this total is **6 or greater**, remove the **bug** and **discard** it from the game. *Otherwise* (total 5 or less), take a **bug spray** from the **Junkyard** (if any remain) and place it on your **motherboard**, and keep the **bug** connected to your **computer**.

OPTION
3

Attempt a Theft

You can try to steal **one** tile from *either* an opponent's **computer** or **workbench**.



STEALING FROM THE WORKBENCH

Select a **hardware tile** in your opponent's **workbench**. **Roll the die**, and **subtract** from your roll the number of **locks** (if any) on your opponent's **motherboard**. If the result is **7 or greater**, the theft is a success (see below). For example, if you roll a 9, and your opponent has 1 lock, your result would be 8 (success).

If the theft failed (result 6 or less), you may **roll again** to try and steal a **different** tile in the **same** opponent's **workbench**. **You may keep repeating theft attempts**, until you have either stolen **one** tile, or you have rolled unsuccessfully at most once for each tile in their **workbench**, at which point your turn ends.

STEALING FROM THE COMPUTER

Select a tile on your opponent's **computer**. You may *not* steal the **motherboard**, a **bug**, any **bugged hardware** (see **Page 12**), or any tile that would cause other tiles to become **loose** (see **Page 5**). That is, you can only steal from the **computer** "outside-in".

Roll the die, and **subtract** from your roll the number of **locks** (if any) on your opponent's **motherboard**. If the result is **7 or greater**, the theft is a success (see below). *Otherwise*, your turn ends. On a failed computer theft, you may *not* make further theft attempts.

Upon a successful theft: Take the stolen tile and place it face-up in your **workbench**.

If your opponent currently has **fewer than 2 locks**, they collect a **lock** from the **Junkyard** and place it on their **motherboard**. **A player may never have more than 2 locks**.



PHASE I: PRE-BUILD (cont'd)

Making Deals

While it's your turn, you may make deals with any other players. Deals may include buying, selling, trading, bargaining, making alliances or threats, and so on. Any kind of deal is allowed, so long as **no rules of the game are violated**. Enforcement of deals and keeping of promises is at the discretion of the players.

As part of deals, players may exchange **JunkBucks**, **hardware** tiles, **locks**, and **bug sprays**. Deals may **not** include items in the **Junkyard** or **JunkBucks** in the **bank**. Players may **not** trade **motherboards**, **software** tiles, **bugs**, or any **bugged hardware** (see **Page 12**). A player may never possess more than **2 locks**.

TRACKING SHEETS

As players buy, sell, and trade tiles, you may find it useful to track the hardware every player (including yourself!) has with a **tracking sheet**. Tracking sheets are optional, and there are no rules about using them. We recommend using one row for each player, and marking in the columns the number of each hardware type that players have.

HARDWARE EXCHANGES

If a player receives a **hardware** tile as part of a deal, they must place it in their **workbench**. It may **not** be connected to their **computer** until **Phase II**.

If a player removes a **hardware** tile from their **computer** as part of a deal, they must move any resulting **loose tiles** to their **workbench**.

COMPUTER JUNKYARD HARDWARE TRACKING SHEET													
PLAYERS													
	ME				/	/	/	/	/	/	/	/	/
	ALLAN	/	/	/	/					/	/	/	/
	JORDAN			/	/			/		/	/	/	/
Hardware Types													
CPU													
HDD													
RAM													
DATABUS													
VIDEO CARD													
MONITOR													
AUDIO CARD													
SPEAKER													
USB CARD													
KEYBOARD													
MOUSE													
PCI													
CD-ROM													
POWER SUPPLY													
SPLITTER													

PHASE II: BUILD THE COMPUTER

With Phase I complete, *it's time to build!* During Phase II, all players will compete *at the same time* to assemble their **computers**. When everyone is ready, the **Sysadmin** starts the 2-minute timer, and players begin building (see below). Once the timer runs out, any player may call "time!", at which point all building must stop. See **Once Time Expires (Page 13)**.

To build your computer, you may do the following in any combination, until time runs out. You may also choose to finish building early.

- Connect, disconnect, and rearrange tiles on your **chassis**. You **cannot** disconnect or rearrange **bugged hardware** (Page 12).
- **Transfer tiles** between your **chassis** and **workbench**.
- **Slide or rotate** your entire **computer** on the **chassis**, *including* any chains of **bugged hardware** (Page 12).
- **Swap a tile** for one of the same type in the **Junkyard** (Page 13).
- **Install** your software tile as a final action (Page 13).



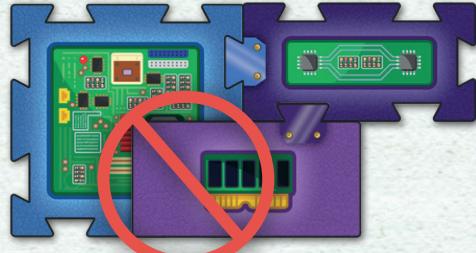
PHASE II: BUILD THE COMPUTER (cont'd)

CONNECTING TILES

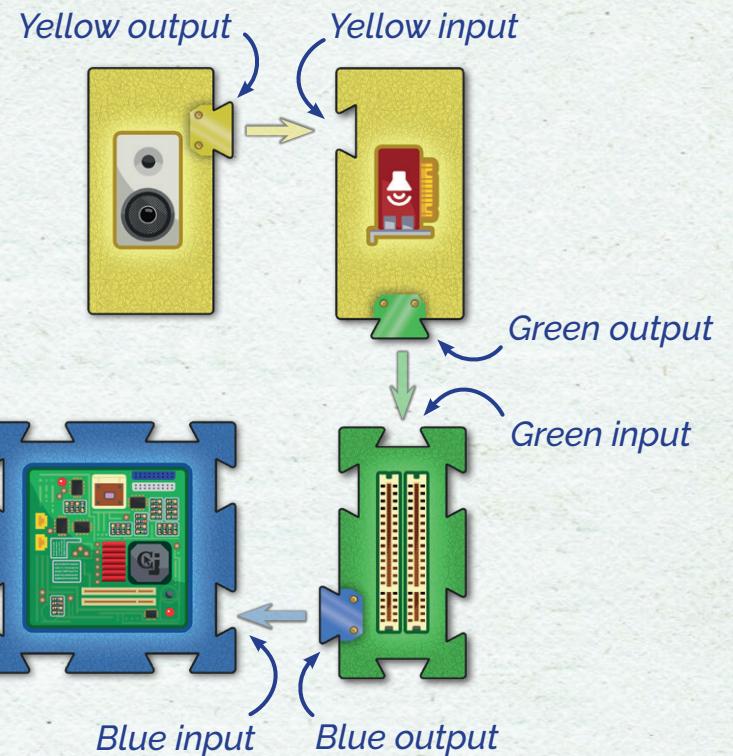
Tiles connect using **input and output ports**. Ports are **color-coded**: if two ports' colors match, they may be connected. The **block diagram** also shows which connections are correct, according to matching colors (see the example on [Page 6](#)).



INCORRECT
Pink output does *not* match orange input.



INCORRECT
Tiles *cannot* overlap each other.



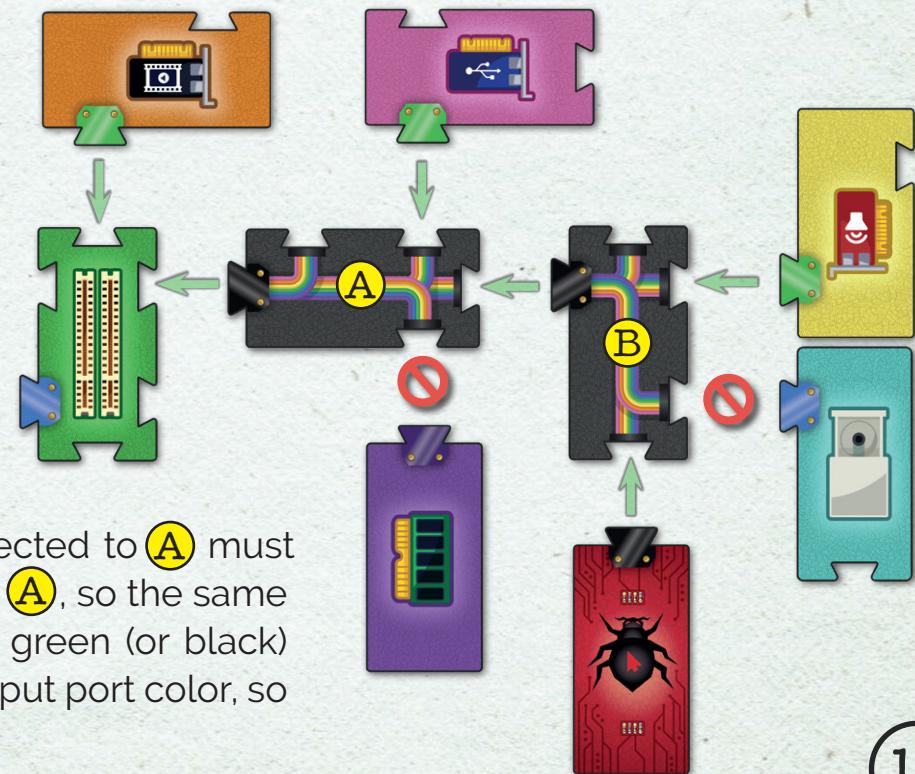
Black ports are unique: they match *any* port color. **Bugs** and **Splitters** are the only tiles with black ports. See below for the correct usage of Splitters. Bugs can be connected to any port during Phase I (see [Page 7](#)), but *cannot* be disconnected while building in Phase II (see [Page 12](#)).

SPLITTERS

Splitters allow **multiple tiles** to connect to **one input port**. Every tile connected to a Splitter must have an **output port** which matches the **color of the tile** to which the Splitter's input connects. Splitters may connect to other Splitters, creating a **chain**. Any number of input ports on a Splitter may remain open throughout the game.

Splitters are optional. For an extra challenge, see if you can win the game without using any!

Example: Splitter **(A)** connects to a PCI, so every tile connected to **(A)** must have a green (or black) output port. Splitter **(B)** is **chained** to **(A)**, so the same rule applies: every tile connected to **(B)** must also have a green (or black) output port. The RAM and CD-ROM tiles have the wrong output port color, so cannot connect to either Splitter.



PHASE II: BUILD THE COMPUTER (cont'd)

BUGGED HARDWARE

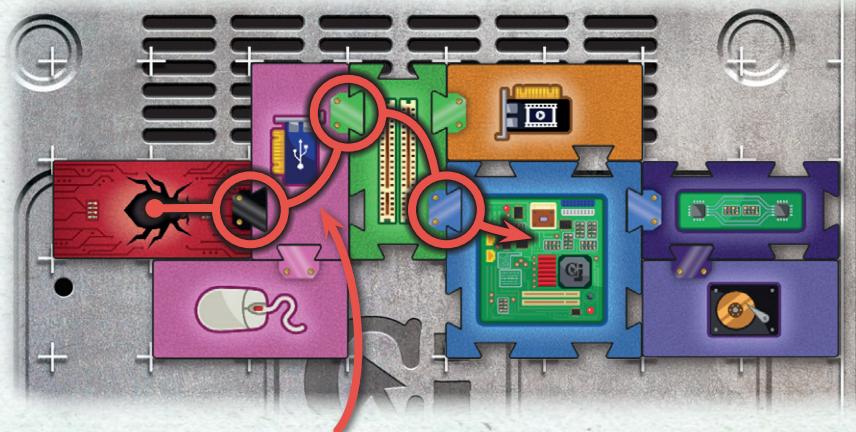
While a **bug** is connected to a **computer**, the chain of tiles *from the bug back to the motherboard* is **bugged**. The tiles in this chain **cannot** be disconnected, rearranged, sold, traded, or stolen. However, any remaining available **input ports** on these tiles may still be used as normal. While building, you **may** slide and rotate your entire **computer**, so long as **bugged hardware** tiles are not disconnected.

The only way to disconnect a **bug** is to **debug** during Phase I (see Page 9). Once a bug is removed, its chain of **bugged hardware** returns to normal.



Note: a **bug** **may** be connected directly to a **motherboard**. In this case, there are no additional bugged hardware tiles; the bug simply cannot be disconnected from the motherboard.

Example: The red line in this diagram follows the chain of **bugged hardware** from the bug to the motherboard. The circled ports are those which cannot be disconnected. In particular, the USB Card and PCI are **bugged**. The other input ports on these tiles may still be used freely; for example, the Video Card and Mouse may be disconnected or rearranged, and additional tiles may be connected.

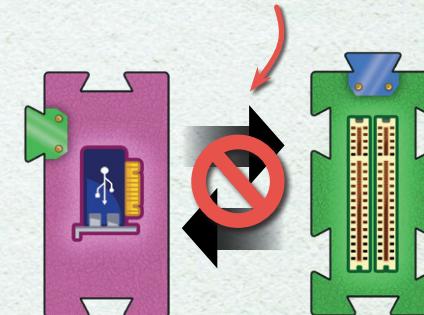
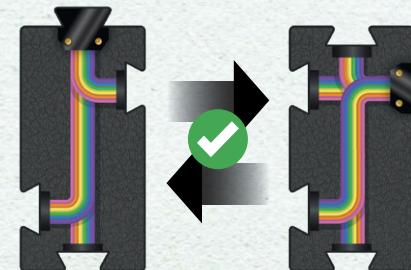
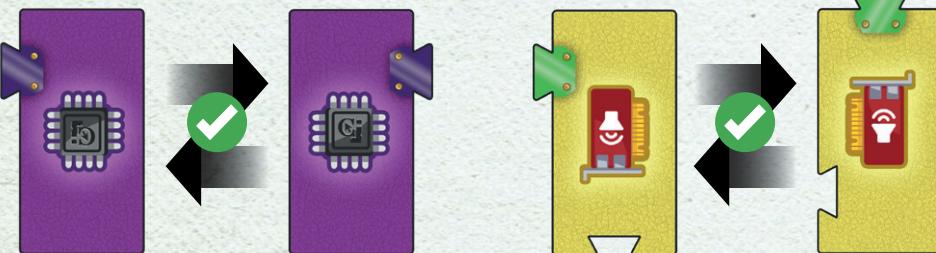


Bugged hardware chain

SWAP TILES WITH THE JUNKYARD

While building, you may **swap** one hardware tile, from your **workbench** or **computer**, with one **tile of the same type** in the **Junkyard**. To perform a swap, **declare "Swap!"** to inform the other players, then place your tile in the **Junkyard**, and collect a tile of the same type from the **Junkyard**. **There is no cost or limit** to the number of swaps you can make before time runs out. **Examples of valid and invalid swaps include:**

Announce "Swap!" each time you perform a swap.



Tiles must be of the same type.

PHASE II: BUILD THE COMPUTER (cont'd)

INSTALLING SOFTWARE

If you believe your **computer** is complete (**see all requirements on Page 14**), connect your **software** tile *directly* to a CD-ROM tile on your computer. *There cannot be a splitter between the CD-ROM and software.*

Once the software is connected, *and* once your **computer** meets the final alignment conditions described below, **announce** “**Installed!**” and wait until Phase II ends. Your computer will be inspected during Phase III. If more than one player declares “**Installed!**” before time runs out, and more than one meets all win conditions, the winner in Phase III will be whoever was the **first** to declare.



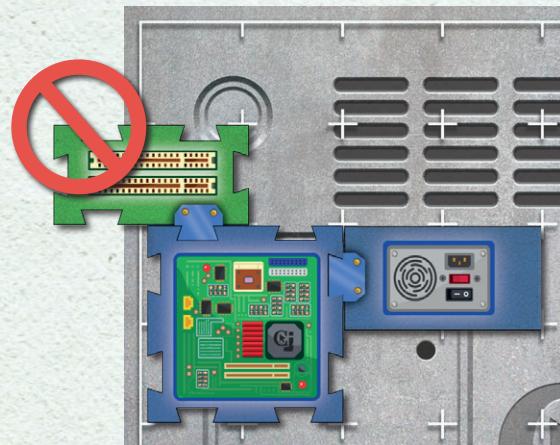
Announce “**Installed!**” once you connect your software.

ONCE TIME EXPIRES

Stop building! Players may make a **final alignment** of their **computer** to the **chassis** grid. If the computer is too large to fit within the grid, disconnect tiles that extend past the edge of the grid as necessary. **Bugged hardware tiles must remain connected** as they were at the beginning of Phase II, even if that requires disconnecting other tiles. **Move all loose tiles** to the **workbench**.

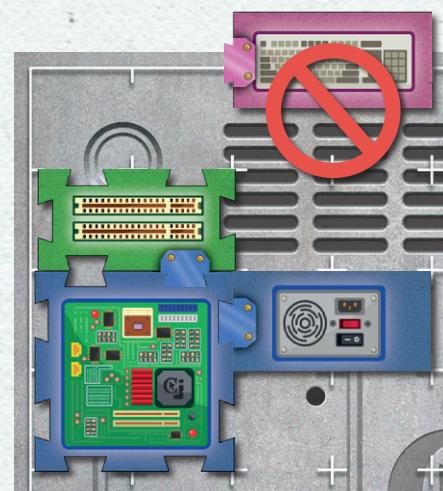
Once all players have made their final alignment, proceed to Phase III.

Players may **not** slide or rearrange their **computers** until the next Phase II.



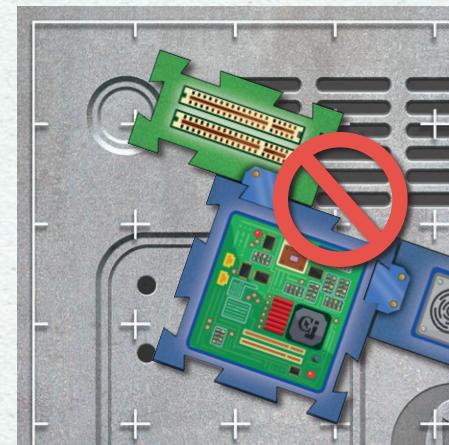
INCORRECT

Tiles cannot extend past the chassis edge.



INCORRECT

Loose tiles must go in your workbench.



INCORRECT

Tiles must be aligned to the grid.

PHASE III: SYSTEM CHECK



You've made it to the end of the round! If any players declared "Installed!" during Phase II, see below to **check if they have won the game**. If more than one player has met the conditions to win, whoever was the **first** to declare "Installed!" is the **winner**.

If no player has won: the **Sysadmin** passes the **timer clockwise**, making the next player the new **Sysadmin**. Return to Phase I to begin the next round. During this time, players may also inspect each other's **computers** to find **build violations** (see next page).

WINNING THE GAME

For a player to **win the game**, their opponents must check that their **computer** meets all of the conditions below. **If any condition is not met**, the player must return their **software tile** to their **workbench**, then address any **build violations** on their computer according to the process on the next page. Gameplay then proceeds as usual.

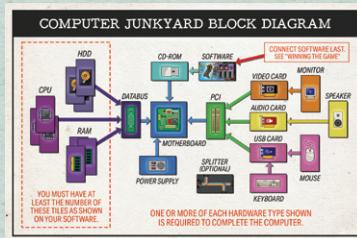
1

The computer includes one (or more) of each of these types of tiles.



2

All tile connections match the block diagram, and any connected Splitters are used correctly (see Page 11).



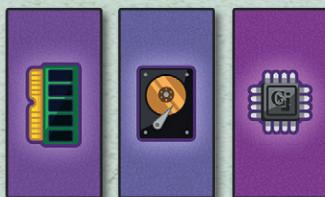
3

The computer has no bug tiles connected.

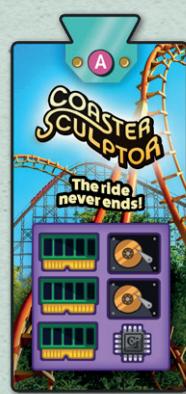


4

The computer includes at least the number of HDDs, CPUs, & RAMs as shown on the software tile.



Example: This software requires at least 3 RAMs, 2 HDDs, and 1 CPU.



5

The software is connected **directly** to a CD-ROM, with no Splitter **in between** them.

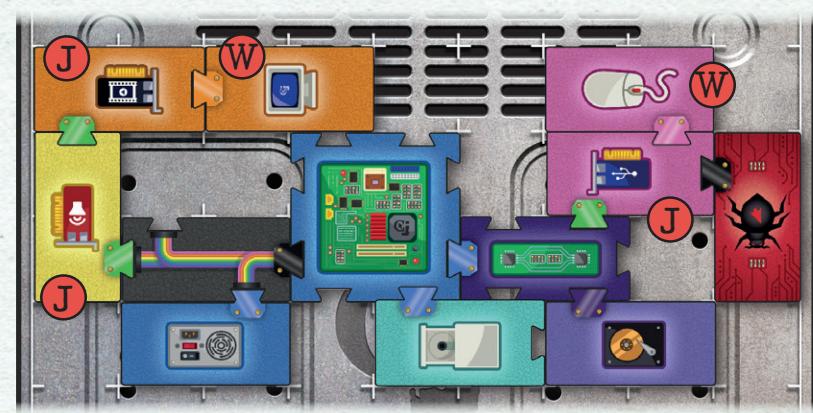


CORRECTING BUILD VIOLATIONS

If at any time during Phase I or Phase III (not during Phase II), you notice a player has a tile connected incorrectly on their **computer**, you *may choose* to declare the error and halt the game. The player must disconnect their **incorrect tile** and return it to the **Junkyard**, then move any resulting **loose tiles** to their **workbench**. If this causes a **bug** to become **loose**, you (the player who noticed the error) may reconnect the bug to any available port of your choice on their **computer**.

After completing these steps, gameplay resumes as normal.

Example: The tiles marked **J** are connected incorrectly, so must be disconnected and moved to the **Junkyard**. The tiles marked **W** will then become **loose**, so must be moved to the player's **workbench**. The **bug** will also become **loose**, so the player who called out the errors may reconnect it to any available port on this computer.

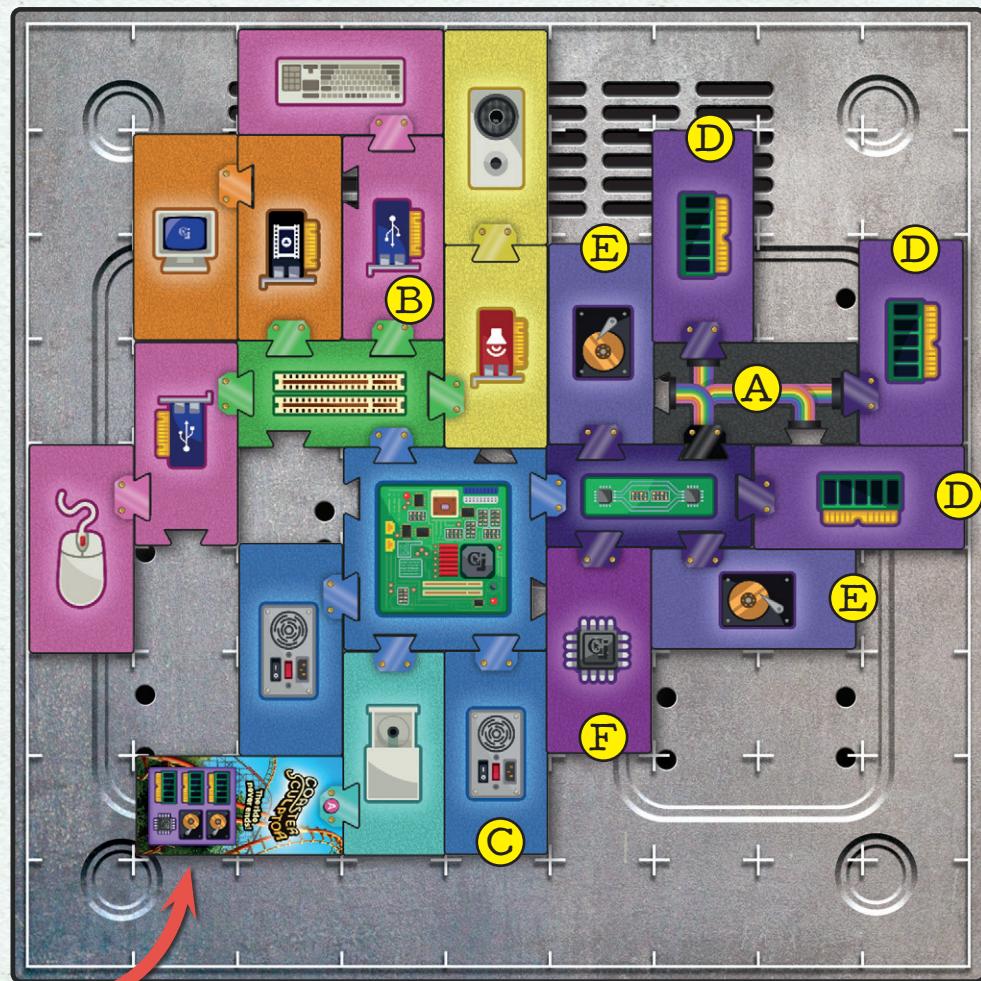
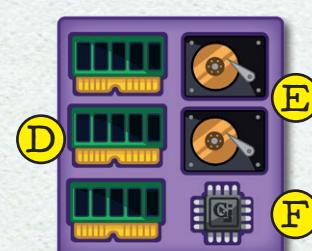


THE FINISHED COMPUTER

Here is an example of a complete computer which meets all required winning conditions.

Note some of the tiles it includes:

- A** An optional Splitter, used to connect two RAMs to the Databus.
- B** A spare USB Card, used to connect the Keyboard.
- C** A spare Power Supply.
- D** 3 RAM tiles, meeting the minimum requirement of the software tile.
- E** 2 HDD tiles, meeting the software requirement.
- F** 1 CPU tile, meeting the software requirement.



GAMEPLAY VARIATIONS

PART-TIME JOB *EASIER*

During Phase I, when a player is *taking an action*, they have a 4th option to simply collect **\$4** from the bank and end their turn.

DEFAULT SETTINGS *EASIER*

During setup, only select **software tiles** from Group **B**, and none from Group **A**. This way, all players will have the same requirements to complete their computers.

COMPACT CHASSIS *HARDER*

Seasoned players can block off the topmost row and rightmost column of their chassis, limiting their build area to an 8×8 grid. All tiles on their computer, including played bugs, must fit inside this smaller grid.

SUPPLY SHORTAGE *HARDER*

During setup, before mixing tiles into the pouch, additionally select and **discard one of each of the following tiles**: CD-ROM, Power Supply, Mouse, Keyboard, Monitor, Speaker. This will mean there is exactly 1 of each of these tile types per player.

SINGLE-PLAYER MODE

The goal of this mode is to build a complete computer as quickly as possible. You will need a stopwatch to track your time. Compete with your friends for the best time!

Collect the following game components: one **chassis**, one **motherboard**, the tile **pouch**, a **block diagram**, all **97 hardware tiles** (do *not* use **bug** tiles), and any one random **software** tile selected *face-down*. To set up, simply mix all 97 **hardware** tiles into the **pouch**. Then, place the **motherboard** somewhere on the **chassis** grid.

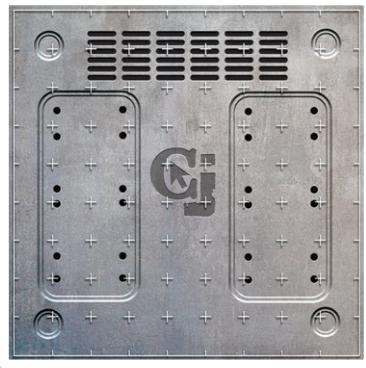
To begin playing, *start your stopwatch*, and immediately flip over your **software** tile to learn the requirements for your **computer**. Then, proceed to build your **computer** by *drawing tiles one at a time* (as quickly and often as you like) from the pouch. Use any tiles you draw at any time; there is no distinction between the **Junkyard** and your **workbench**.

The moment you believe you have completed your computer, install your **software** and *stop your stopwatch*. Inspect your computer to make sure it satisfies all win conditions on **Page 14**. If you meet all conditions, then your "score" is the time on the stopwatch. If any condition is not met, the run is invalid.

ANATOMY OF A COMPUTER

The Real Tech Behind the Tiles

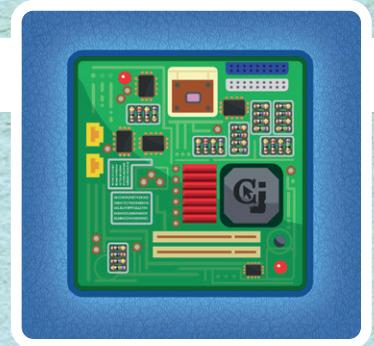
Computer Junkyard features a variety of components found in real computers. In this section, we summarize the real functionality of these components, and how we chose to represent them in the game.



CHASSIS

A computer's chassis is the case that houses and protects its internal components. In *Computer Junkyard*, it acts as the base where the motherboard and all other tiles connect together. This includes components which usually (especially for a vintage computer) lie outside a real chassis, namely the mouse, keyboard, speakers, and monitor. However, for some modern devices like laptops and smartphones, these components are sometimes integrated right into the chassis!

MOTHERBOARD



A computer's motherboard is a large circuit board that holds and connects some of the most important central hardware, such as the CPU, RAM, and expansion slots. In *Computer Junkyard* we represent this by having the motherboard tile be the hub of the computer you assemble.

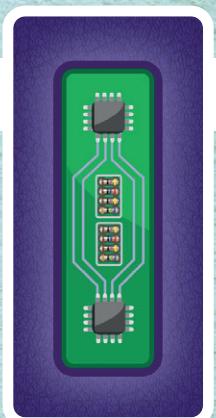


POWER SUPPLY

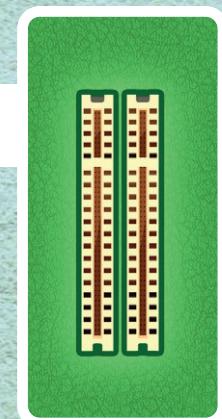
A power supply takes in power from an electrical network, and delivers it in a regulated manner to all of the computer's hardware. Power supplies generate a lot of heat, so fans are used to cool the system by expelling heat through exhaust vents in the chassis. Other components like the CPU and GPU produce a lot of heat as well, and can require more elaborate cooling systems.

Rather than having an internal power supply, many modern devices like laptops have an external "power brick" built into their charging cable, which supplies electricity to an internal battery.

DATABUS



A databus (or simply "bus") is a collection of wires used to exchange data between components. A real computer contains many databases of various types and sizes. In *Computer Junkyard*, we summarize this idea with one databus tile that connects some of the more data-hungry components, namely the CPU, RAM, and HDD.



PERIPHERAL COMPONENT INTERCONNECT (PCI)

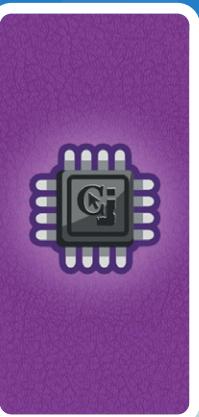
The PCI provides *expansion slots* used to connect additional components, or *peripherals*, to the computer. *Expansion cards* act as the interface between these slots and the peripheral hardware; in the game, these are the Audio, Video, and USB Card tiles. Technically, "PCI" refers to a specific expansion card protocol which was common in older computers, but has since been supplanted by more compact and efficient technologies.

ANATOMY OF A COMPUTER

The Real Tech Behind the Tiles

CENTRAL PROCESSING UNIT (CPU)

The CPU is the command center of the computer. It controls the computer's core functions, coordinates and directs peripheral hardware, performs arithmetic calculations, manages data flow, runs the operating system and applications, and more. A CPU's speed is measured by how fast it can run instructions; CPUs today run thousands of times faster than those from decades ago. Modern CPUs also feature multiple *cores*, allowing them to perform multiple tasks at the same time.



RANDOM ACCESS MEMORY (RAM)



RAM is high-speed data storage used by the CPU to temporarily hold data it is currently working with, such as data associated with currently running applications or open documents. RAM is "volatile" memory, meaning the stored data is lost when the computer loses power.

If the computer tries to store too much in RAM at once, it must overflow to the much slower long-term data storage (see HDD below). Historically, RAM storage limits were often very small, and adding more RAM was an expensive upgrade to significantly improve a computer's performance.

HARD DISK DRIVE (HDD)



Computers need stable long-term data storage which is "non-volatile," meaning the data is retained even when the computer is turned off. For decades, the most common non-volatile storage technology has been the hard disk drive. An HDD uses a moving arm to read and write data onto spinning metal disks, with data encoded magnetically on the surface of the disks.

In recent years, *solid-state drives* have surpassed HDDs in popularity as a long-term storage solution, as they are faster and more compact than HDDs, and have no moving parts.



AUDIO CARD & SPEAKER

Speakers play audio through a very basic mechanism: they simply vibrate the air, so our ears can receive the vibrations and interpret them as sound. Speakers often come in pairs to produce *stereo sound*, which allows our brain to sense the "direction" of the sound.

An audio card has specialized circuitry, which enables it to take digital data representing sound, and convert it into an electrical signal to drive the speaker's vibrations.

UNIVERSAL SERIAL Bus (USB) KEYBOARD & MOUSE

USB is a standardized hardware and data transfer protocol that allows a computer to communicate with a wide variety of peripheral devices, such as keyboards, mice, webcams, game controllers, drawing tablets, flash drives, and MIDI instruments. USB-connected devices exchange data *serially* (in a sequence), which is simpler and cheaper than more advanced data protocols.



While most motherboards feature built-in USB ports, a USB Card can be connected to the PCI to add even more ports.

ANATOMY OF A COMPUTER

The Real Tech Behind the Tiles



VIDEO CARD & MONITOR

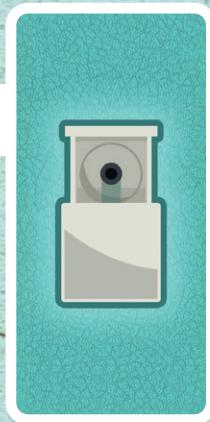
Monitors have been an essential way we interact with computers for decades. The monitor tile in *Computer Junkyard* depicts a classic *CRT* display. These were large and heavy monitors, containing an electron beam that was directed by magnetic fields to hit and illuminate the curved screen at the front.

Video cards are designed to move a lot of data at high speeds, to render graphics on a monitor at high enough frame rates for practical use. Modern video cards, also called GPUS, can often be the most expensive component in a computer. They are specialized for demanding computations with large amounts of data, to support tasks like high-fidelity gaming, 3D rendering, and even machine learning for artificial intelligence.



SPLITTER

A splitter cable allows multiple devices to connect to one port. Depending on the type of port being split and devices being connected, a splitter will use a different protocol to divide the data being exchanged. In *Computer Junkyard* we depict splitters as rainbow ribbon cables, to emphasize their compatibility with every port color in the game. In reality, ribbon cables are only used as splitters in specific circumstances.



COMPACT DISC READ-ONLY MEMORY (CD-ROM)

CD-ROM is a data storage format in which data is encoded as tiny variations on the surface of a metal disc. This technology was first used for storing music, replacing cassettes and vinyl records; its use then expanded to general data storage, and was popular for the distribution of computer software and games. Technically, "CD-ROM" refers to the disc itself, while the hardware that reads the disc is a *CD-ROM drive*, or more generally an *optical drive*. An optical drive shines a laser at a disc to read the data encoded on the disc's surface.



SOFTWARE

Every application has a *specification* of the hardware it requires to run smoothly, notably the power of the CPU and the needed hard drive and RAM storage space. Modern computers are powerful enough that these requirements typically only matter for the most intensive applications, such as high-fidelity video games, complex simulations, and video editors. In contrast, computers decades ago had tight performance limits; users had to consider what software they wanted to run when buying or building a computer. We revive this experience in the gameplay of *Computer Junkyard*, where each player has unique requirements to run their software.



COMPUTER BUG

Bugs are defects or mistakes in hardware or software, which can cause a computer to malfunction or crash. *Computer Junkyard* focuses on hardware bugs, as gameplay involves attaching bugs to hardware tiles. A big responsibility of computer programmers and engineers is finding and fixing bugs. However, because computers are such complex systems, finding all bugs is generally impossible. So, these systems are also designed to be resilient to moments of failure.

Famously, some hardware bugs in the past have been caused by actual insects getting caught inside computer circuitry! This inspired the *bug spray* mechanic of *Computer Junkyard*.

FREQUENTLY ASKED QUESTIONS

Q: Can I make a deal where I try to debug someone else's computer on my turn?

A: No, during your turn you may only try to remove bugs from your own computer. However, one way you can help another player debug is by making a deal where you offer them bug sprays you have! **See Page 10.**

Q: I want to buy 2 of the same type of tile from the Junkyard. Are they both the same price?

A: No. Each purchase and sale is handled individually. That means when you buy 2 of the same tile, the second will (usually) be more expensive than the first. **See Page 8.**

Q: If I draw a bug, can I play it on myself if I want to?

A: Yes, but you are never required to do so. In particular, if the bug cannot connect to any opponent's computer, you can choose to play the bug on yourself or discard it. **See Page 7.**

Q: Can I connect more than one of the same tile type to my computer?

A: Yes, and it can be strategic to do so! You may connect any number of each type of tile to your computer, as long as all port connections are correct according to the block diagram.

CREDITS & THANKS

GAME CREATORS

Allan & Jared Pincus

ART & GRAPHICS

Jordan Pincus

"HUMAN RESOURCES"

Lauren Harris-Pincus

GAME DEVELOPMENT SUPPORT

Tim Checko
Mike O'Connell
Ryan Simpkins

SPECIAL THANKS (alphabetically)

Aiden Blow, Samantha & Franklin Clark, VFC,
The Counihans (Tim, Jack & Mason), Richard Decker,
Dustin Droz, Mike Emerson, Paul Fornale,
Aubrey Odom, Rafal Polinski, Jon Trudel, Owen Zila

MORE INFORMATION

Visit our website in order to:

- Watch our How-To-Play video!
- Leave a comment or question!
- Order replacement game components!



DreamEggGames.com