

# C++ OOP – Regular Exam – 20 April 2024

Submit your zip file here: <https://judge.softuni.org/Contests/Compete/Index/4740>

## 3. Clothes Robot

You must finalize the logic of a dress producing robot. The robot can execute the following commands:

- **"cut ID t m"** - which has three parameters: the **ID** of the piece to be created, the amount of time **t** which is necessary for the operation to be completed, and the amount of material **m** to be used. This command creates the corresponding piece in the list of pieces.
- **"sew ID t ID1 ID2 .. IDn"** - which has variable number of parameters. The **ID** is the identifier of the piece to be created, the **t** is the time to do the sew operation. The rest of the parameters are a list of all IDs **ID1 ID2 .. IDn** of the pieces to be sewn together to create the resulting **ID**.
- **"iron ID t"** - which has the identified **ID** of the piece to be ironed and time **t** to iron it
- **"pack"** - which has no parameters and indicates the dress must be packed and shipped. This is the last command of each dress, and it produces resulting output. Check below and in the examples for detailed explanation.
- **"end"** - which terminates the program functionality, without printing anything: the program just stops there.

The robot receives a list of commands, which describe various dresses to be produced. The robot first reads all the commands in its memory, and after that executes them in their order of appearance.

During the processing, the robot must take care to handle **only one possible error: incorrect ID**. This can happen in one of the following cases:

1. The **ID** of the item to be created by the commands **cut** and **sew** already exists in the storage. In this case the commands fail because they cannot create another item with the same ID.
2. The **ID**, which the command **iron** receives, or one or more of the **ID1 ID2 .. IDn**, which the command **sew** receives to act upon, are not present in the storage. In this case the commands fail because they cannot execute.

**Please note:** the command **sew** could receive more than one invalid **IDs**. In this case we abort on the first in the order of receiving, if the command **"sew something 17 af1 existingId af2"** both **af1** and **af2** are invalid, we will abort on **af1**: this will be the one shown as **"{the invalid ID}"**.

If there's an error the robot must print the following message:

Aborting on {command name} due to invalid ID "{the invalid ID}". Lost pieces {comma-separated list of alphabetical sorted all IDs that are currently in processing}, lost material {the total quantity of all used material}, wasted time {the total amount of all time, used so far}.

**Please note:** When a process aborts, it clears out all storage and processing, until the corresponding **pack** command. See the second example below for more details.

In case the robot receives the **pack** command, it prints out the successful output of all processing:

Packing and shipping new order: "{comma-separated list of alphabetical sorted all IDs that are currently in processing}". Production material: {the total quantity of all used material}, production time: {the total amount of all time, used so far, without the time of the errored operation}.

As usual, In the skeleton you'll find partial definition and implementation of the program. Your task is to finish it.

**Please note:** The output will always be correct, except that some IDs will be mistaken to test the functionality.

The ideal solution zip file would include your implementation of **Command.cpp**, **SewCommand.cpp**, **PackCommand.h**, and **IronCommand.h**, but you're free to choose a different strategy as well.

## Example Input / Output

Input	Explanation
cut sleeveR 5 8 cut sleeveL 6 8 cut base 12 15	Creates three pieces with corresponding IDs: 1. Right sleeve, for 5 minutes, using 8 materials, with ID sleeveR; 2. Left sleeve, for 6 minutes, using 8 materials, with ID sleeveL; 3. Base of the jacket, for 12 minutes, using 15 materials, with ID base.
sew light-jacket 7 sleeveL base sleeveR	Sews together all three elements, using 7 minutes, and creating the light-jacket piece from pieces sleeveL base and sleeveR.
iron light-jacket 5	Irons the light-jacket piece for 5 minutes.
pack	Packs and delivers the light-jacket piece. Outputs: Packing and shipping new order: <b>"light-jacket"</b> . Production material: <b>31</b> , production time: <b>35</b> . Explanation of totals: Total time: $5 + 6 + 12 + 7 + 5 = 35$ for each consecutive operation. Total materials: $8 + 8 + 15 = 31$
end	

## Example Input / Output

Input	Output	Explanation
cut sleeveR 5 8 cut sleeveL 6 8 cut base 12 15		Creates three pieces with corresponding IDs: 1. Right sleeve, for 5 minutes, using 8 materials, with ID sleeveR; 2. Left sleeve, for 6 minutes, using 8 materials, with ID sleeveL; 3. Base of the jacket, for 12 minutes, using 15 materials, with ID base.
sew light-jacket 7 sleeve base sleeveR	Aborting on sew due to invalid ID "sleeve". Lost pieces <b>base, sleeveL, sleeveR</b> , lost material <b>31</b> , wasted time <b>23</b> .	Errors out, because there's no sleeve element in the storage. Explanation of totals: Total time: $5+6+12=23$ for each consecutive operation so far, without the errored operation. Total materials: $8+8+15 = 31$ . The process aborts and the following iron command and pack command are not executed. The processing continues after them.
iron light-jacket 5		This command never executes: the algorithm skips all commands until after the incoming pack command.

pack		This command never executes.
cut I1 3 5 cut I2 4 7 sew J1 2 I1 I2 iron J1 2		Creates the elements I1, I2 for total of 7 (3+4) minutes and 12 (5+7) materials. Sews them into J1 for 2 minutes. Irons J1 for 2 minutes. The total cost of J1 is 3+4+2+2=11 minutes and 5+7=12 materials.
cut I3 5 8 sew J2 3 J1 I3 cut I3 2 2		Creates new element I3 for 4 minutes and 8 materials. Creates J2 by sewing up J1 and I3 for 3 minutes. Creates new element I3 for 2 minutes using 2 materials. <b>Please note:</b> there's new I3, which is created OK, since the previous I3 is already used for production!
pack	Packing and shipping new order: "I3, J2". Production material: 22, production time: 21.	Packs and sends the everything, prints the output. <b>Please note:</b> The wasted materials and time from the previous "Aborting" message are not included, as the <b>sew</b> command above aborted, throwing out everything from the existing storage!
end		