# Warcraft

### Intro

Your task is to write a program that represents parts from the famous computer game – “**Warcraft**”.  
Three famous heroes of the game take place into the scene.

Archmage – human wizard

DeathKnight – undead knight

DrawRanger – half elf, half undead ranger

You will simulate those heroes casting basic and ultimate spells.

### Provided input

You are provided with folder called **Skeleton**, which contains:  
- A header file that contains the most important information for the task.

- A Basic Skeleton to guide you in the initial direction. You don’t need to strictly follow it.

You are also provided with folder called **tests**, which contains all of the tests that your program should pass.

- There are a total of 13 tests. 13 inputs and 13 outputs. The 3 of the tests are copy of the ones described in this document. The other 10 are unique.

**Task description**

Your task is to provide to **provide an implementation**, which with the given standard input will produce the correct standard output (detailed description below).

Each hero has the following attributes:

* name – name of the character;
* maxMaxa – the character mana pool for casting spells. (If you don’t know what ‘mana’ is – think of it as a currency required to cast a spell).
* baseManaRegenRate – tell you how much mana points your character restores when an ActionType::REGENERATE\_MANA is performed. Keep in mind that your character can **NOT** have more mana points than his “maxMaxa”. Your character can restore mana points **UP** to his “maxMaxa”.

**NOTE**: The **Archmage** class has a special bonus attribute: “manaRegenModifier”, which scales up his mana regeneration (multiplies baseManaRegenRate to manaRegenModifier) each time the character performs an ActionType::REGENERATE\_MANA.

Each character has his unique BASIC and ULTIMATE spells that are already predefined.

**#define** MAX\_SPELL\_NAME\_SIZE 20

**typedef** **struct** {

**char** name[MAX\_SPELL\_NAME\_SIZE];

**int** manaCost;

} Spell;

You are given the main() function, which read the data for the Archmange, DeathKnight and DrawRanger. It is your task to implement them using this input.

Next a single integer (N) is read from the standard input.

* The next N whitespace separated integer are special ActionType commands;

**enum** ActionType {

*CAST\_BASIC\_SPELL*, *CAST\_ULTIMATE\_SPELL*, *REGENERATE\_MANA*

};

1. “0” or ActionType::CAST\_BASIC\_SPELL command – all heroes should **TRY** to casts their BASIS spells (if they have enough mana points);
2. “1” or ActionType::CAST\_ ULTIMATE \_SPELL command – all heroes should **TRY** to casts their ULTIMATE spells (if they have enough mana points);
3. “2” or ActionType::REGENERATE\_MANA command – all heroes should use their ability to regenerate mana;

### Console output

After each ActionType::CAST\_BASIC\_SPELL or ActionType::CAST\_ULTIMATE\_SPELL each hero should **print to the console** a result of his actions.

1. For successful cast you should print: ‘spell name’ casted for ‘spell mana’ followed by a **newline**.
2. For unsuccessful cast you should print: ‘spell name’ – not enough mana to cast ‘spell name’ followed by a **newline**.

Note: ActionType::REGENERATE\_MANA does **NOT** print any result to the console.

**Special hero abilities**:

* Archmage – if SpellType::ULTIMATE is successfully casted the Archmage gets **immediately** a free ActionType::REGENERATE\_MANA.
* DeathKnight – if SpellType::ULTIMATE is successfully casted the DeathKnight gets **immediately** a free ActionType::CAST\_BASIC\_SPELL.   
  **Important note**: on the free basic cast spell you should print to the console – ‘spell name’ casted for 0 mana (cast the spell for 0 mana).
* DrawRanger – if SpellType::BASIC is successfully casted the DrawRanger gets **immediately** a free ActionType::CAST\_BASIC\_SPELL.   
  **Important note**: on the free basic cast spell you should print to the console – ‘spell name’ casted for 0 mana (cast the spell for 0 mana).

### Restrictions

Code should compile under the C 11 standard.

Code should not use any non-standard extensions.

### Examples

|  |  |
| --- | --- |
| **Input** | **Output** |
| Archmage 480 80 2  DeathKnight 420 70  DrawRanger 360 60  2  0 1 | Archmage casted Water Elemental for 120 mana  DeathKnight casted Death Coil for 75 mana  DrawRanger casted Silence for 90 mana  DrawRanger casted Silence for 0 mana  Archmage casted Mass Teleport for 180 mana  DeathKnight casted Animate Dead for 200 mana  DeathKnight casted Death Coil for 0 mana  DrawRanger casted Charm for 150 mana |
| Values 180 50 3  Are-not 220 80  Hardcoded 160 90  2  1 1 | Values casted Mass Teleport for 180 mana  Are-not casted Animate Dead for 200 mana  Are-not casted Death Coil for 0 mana  Hardcoded casted Charm for 150 mana  Values - not enough mana to cast Mass Teleport  Are-not - not enough mana to cast Animate Dead  Hardcoded - not enough mana to cast Charm |
| ConjurusRex 280 80 1  Arthas 320 40  Sylvanas 160 50  4  1 2 2 1 | ConjurusRex casted Mass Teleport for 180 mana  Arthas casted Animate Dead for 200 mana  Arthas casted Death Coil for 0 mana  Sylvanas casted Charm for 150 mana  ConjurusRex casted Mass Teleport for 180 mana  Arthas casted Animate Dead for 200 mana  Arthas casted Death Coil for 0 mana  Sylvanas - not enough mana to cast Charm |