

C++ Helper

در این پروژه می‌خواهیم برنامه‌ای برای آسان‌سازی کار با زبان سی‌پلاس‌پلاس بنویسیم. برنامه کامپایل‌شده شما باید به صورت زیر قابل استفاده باشد. در لینوکس یا مک:

```
$ ./helper format hello_world.cpp
```

```
$ ./helper create-by-template sample_program.cpp
```

```
$ ./helper replace test_file.cpp -r int, long -r float, double -r std::endl,
'\n'
```

یا در ویندوز:

```
C:\> helper.exe format hello_world.cpp
```

```
C:\> helper.exe create-by-template sample_program.cpp
```

```
C:\> helper.exe replace test_file.cpp -r int, long -r float, double -r
std::endl, '\n'
```

برنامه‌ای که می‌نویسید، باید قابلیت‌های زیر را داشته باشد:

- برنامه باید دستورات را به صورت **command-line argument** ورودی بگیرد. (۱۵ نمره)
- دستور `format` در خط فرمان، بدین معناست که برنامه باید فایل ورودی داده‌شده را با قوانینی که در ادامه توضیح داده می‌شود، زیباسازی کند و در نهایت گزارشی (که در ادامه به توضیح آن پرداخته می‌شود) از فایل در خروجی چاپ کند.
- دستور `create-by-template` در خط فرمان، بدین معناست که برنامه باید یک فایل ساده سی‌پلاس‌پلاس با کمک **قالبی** که در ادامه آورده شده، ایجاد نماید و در فایلی با نام **ورودی داده‌شده** ذخیره کند. (۱۵ نمره)
- دستور `replace` در خط فرمان، بدین معناست که برنامه باید یک‌سری از کلمات ورودی داده‌شده را با کلمات خواسته‌شده جایگزین کند. (۳۰ نمره)

قوانین زیباسازی

- هر ۸ کاراکتر فاصله (space) که پشت سر هم آمده باشند را به یک کاراکتر tab تبدیل کنید. (۱۵ نمره)
- کامنت‌هایی که به صورت `/* text */` در فایل ورودی وجود دارند، به کامنت با شکل `// text` تبدیل کنید. (۲۰ نمره)

گزارش نهایی دستور Format

دستور فرمت باید پس از زیباسازی فایل، گزارشی را شامل موارد زیر تولید کند و در خروجی نمایش دهد:

- تعداد خط‌های فایل ورودی. (۵ نمره)
- ماکزیمم تعداد خط‌های توابع در فایل. (۲۰ نمره)

دستور Replace

برای دستور replace در نظر داشته‌باشید که جلوی هر `r` - یک جفت کلمه داده می‌شود که باید کلمه اول را با کلمه دوم جایگزین کرد. برای نگهداری کلمات موردنظر و کلمات جایگزین آن‌ها، حتماً از کلاس `map` کتابخانه استاندارد STL استفاده کنید، در غیر این صورت از نمره شما کسر خواهد شد.

تولید فایل سی‌پلاس‌پلاس با قالب آماده

بسیاری از برنامه‌نویسان برای زبان‌های مختلفی که استفاده می‌کنند، قالبی آماده دارند و هنگام شروع پروژه جدید از آن‌ها استفاده می‌کنند تا هر دفعه مجبور به پیاده‌سازی مجدد یک فایل نباشند. به همین منظور می‌خواهیم وقتی دستور `create-by-template` وارد می‌شود، فایلی ساخته شود و کد آماده نوشته‌شده در پیوند، در آن نوشته شود.

نکات

در طول پیاده‌سازی پروژه به موارد زیر توجه داشته‌باشید:

- هر تابعی خارج از کلاس نمره منفی دارد (به جز تابع `!main`).
- ساختار برنامه باید تماماً براساس اصول `object-oriented` باشد.

- تعریف هر کلاس باید در یک فایل **.h** و پیاده‌سازی آن در یک فایل **.cpp** مجزا باشد. در انتها همه را یک فایل زیپ کرده و آپلود کنید (تصحیح پروژه به صورت دستی خواهد بود و نمره سیستم فقط به این معنی است که شما آپلود کرده‌اید).

- اگر برنامه شما به علت خطای کامپایلر، کامپایل نشود نیمی از نمره را از دست می‌دهید.

نمره این پروژه از ۱۰۰ است و ۲۰ نمره آن امتیازی است.

مثال‌ها

فایل `hello_world.cpp` را در مسیر `/home/ap` به صورت زیر در نظر بگیرید:

```
1  /*
2  Copyright 2018 Amirkabir University of Technology.
3
4  Licensed under the Apache License, Version 2.0 (the "License");
5  you may not use this file except in compliance with the License.
6  You may obtain a copy of the License at
7
8      http://www.apache.org/licenses/LICENSE-2.0
9
10 Unless required by applicable law or agreed to in writing, software
11 distributed under the License is distributed on an "AS IS" BASIS,
12 WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
13 See the License for the specific language governing permissions and
14 limitations under the License.
15 */
16
17 #include <iostream>
18
19 void simple_func();
20
21 int main() {
22     simple_func();
23     return 0;
24 }
25
26 void simple_func() {
```

```

27         // Just a simple cout!
28         std::cout << "Hello World!" << std::endl;
29         std::cout << "Another msg" << std::endl;
30         float magic_number = 2.8;
31         std::cout << "Num: " << magic_number << '\n';
32     }

```

نمونه ۱

```
$ ./helper format /home/ap/hello_world.cpp
```

اجرای برنامه با دستور بالا، خروجی زیر را تولید می‌کند:

Lines: 32

Max Func Length: 5

همچنین فایل hello_world.cpp پس از اجرای برنامه، به صورت زیر تغییر می‌کند:

```

1  //Copyright 2018 Amirkabir University of Technology.
2  //
3  //Licensed under the Apache License, Version 2.0 (the "License");
4  //you may not use this file except in compliance with the License.
5  //You may obtain a copy of the License at
6  //
7  //    http://www.apache.org/licenses/LICENSE-2.0
8  //
9  //Unless required by applicable law or agreed to in writing, software
10 //distributed under the License is distributed on an "AS IS" BASIS,
11 //WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or //impli
12 //See the License for the specific language governing permissions and
13 //limitations under the License.
14
15 #include <iostream>
16
17 void simple_func();
18
19 int main() {

```

```

20     simple_func();
21     return 0;
22 }
23
24 void simple_func() {
25     // Just a simple cout!
26     std::cout << "Hello World!" << std::endl;
27     std::cout << "Another msg" << std::endl;
28     float magic_number = 2.8;
29     std::cout << "Num: " << magic_number << '\n';
30 }

```

نمونه ۲

```
$ ./helper replace hello_world.cpp -r magic_number, num -r float, double -r
std::endl, '\n' -r simple_func, func
```

با اجرا دستور بالا، فایل hello_world.cpp به صورت زیر تغییر می‌یابد:

```

1  /*
2  Copyright 2018 Amirkabir University of Technology.
3
4  Licensed under the Apache License, Version 2.0 (the "License");
5  you may not use this file except in compliance with the License.
6  You may obtain a copy of the License at
7
8      http://www.apache.org/licenses/LICENSE-2.0
9
10 Unless required by applicable law or agreed to in writing, software
11 distributed under the License is distributed on an "AS IS" BASIS,
12 WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
13 See the License for the specific language governing permissions and
14 limitations under the License.
15 */
16
17 #include <iostream>
18
19 void func();

```

```

20
21 int main() {
22     func();
23     return 0;
24 }
25
26 void func() {
27     // Just a simple cout!
28     std::cout << "Hello World!" << '\n';
29     std::cout << "Another msg" << '\n';
30     double num = 2.8;
31     std::cout << "Num: " << num << '\n';
32 }

```

نمونه ۳

```
$ ./helper create-by-template /home/ap/sample.cpp
```

با اجرای دستور بالا، فایلی با اسم *sample.cpp* در مسیر */home/ap* تولید می‌شود که به صورت زیر است:

```

1  #include <iostream>
2  #include <vector>
3  #include <string>
4
5  using namespace std;
6
7  typedef vector<int> ints
8  typedef vector<string> strings
9
10 int main() {
11     return 0;
12 }

```