باسمه تعالی

سوال 1 تمرین سری 6:

الف:

```
#include <string.h>

#include <stdio.h>

#include <stdlib.h>

void fun(char** str_ref)-------→  void fun(char * str_ref)
```

چون در قسمت main فقط اشاره گر به کاراکتر داریم (نه اشاره گر به اشاره گر) و اینکه با توجه به نیاز مسئله که کارمان با اشاره گر به کاراکتر راه میوفتد پس در این قسمت هم ورودی تابع ما فقط باید اشاره گر به کاراکتر باشد و در غیر اینصورت ورودی که در فراخوانی به تابع میدهیم با ورودی که در تعریف تابع استفاده کردیم هم خوانی ندارد. به عبارت دیگه انگار که یک آرایه یک بعدی از کاراکتر داریم و همین جواب گوی مسئله هست ولی در تعریف ورودی تابع آرایه دوبعدی تعریف کرده ایم که این تفاوت باعث ایجاد warning میشود.

```
{

   str_ref++;

}

int main()

{

   char *str = (void *)malloc(100*sizeof(int));----→ char *str = (void *)malloc(100*sizeof(char));
```

چون اشاره گر ما از جنس کاراکتر هست پس باید هر خانه ی حافظه ای که به آن اختصاص میدهیم هم، فضایی به اندازه ی یک خانه ی کاراکتر داشته باشد. مثلا چون رودی های ما یک رشته است که درواقع آرایه ای از کاراکتر هست و مثلا اگر هر کاراکتر 2 بایت فضا بخواهد ما باید این آرایه را با خانه های دوبایتی که برابر حافظه لازم برای ذخیره یک کاراکتر هست پرکنیم.

```
   strcpy(str, "AUTprincipalprogramming");

   fun(str);

   puts(str);


   return 0;

}
```

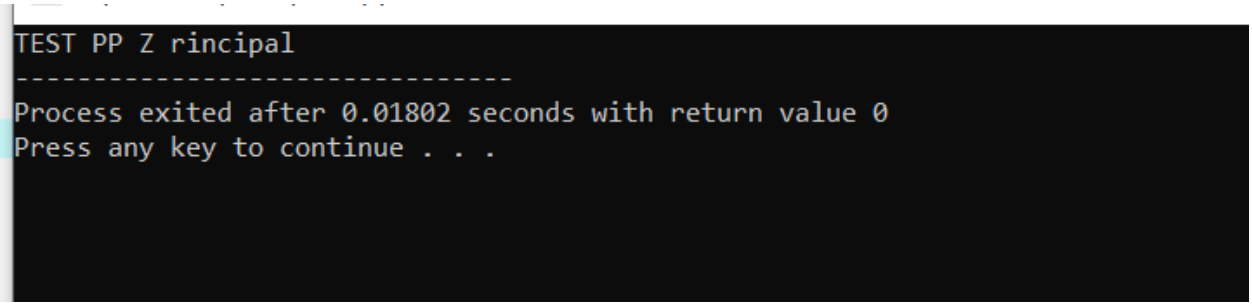
از آنجایی که هدف از تابع fun این بوده که اشاره گر یک واحد به جلو رود پس باید این تغییر مکانی که اشاره گر به آن اشاره میکند یک جور سیو شود یا به عنوان خروجی تابع برگردانده شود.

پس کد نهایی میشود:

```
char *fun(char *str_ref)
{
        str_ref++;
        return str_ref;
}
int main()
{
   char *str = malloc(100*sizeof(char));
   strcpy(str, "AUTprincipalprogramming");
   str=fun(str);
   puts(str);
   return 0;
}
```

ب:

خروجی کد داده شده به شکل زیر می باشد:



دلیل : خروجی هر printf را جداگانه بررسی میکنیم:

1 - printf("%s ", **++cpp);

همانطور که در کد مشخص است cpp** به خانه ی 0 از cp اشاره میکند. خانه ی 0 از cp نیزمقدارc+3 دارد. خود c به تنهایی به ستون 0 از آرایه دوبعدی که تشکیل داده (اشاره گر به آرایه ای از کارکترها) اشاره میکند پسc+3 به ستون سوم که شامل عبارت "QUIZ" می باشد اشاره میکند. در نتیجه حالا که cpp++** داریم پس به خانه اول از cp که مقدار c+2 دارد اشاره میکند که c+2 هم به ستون دوم یعنی"TEST" اشاره میکند. پس خرجی printf اول عبارت "TEST" می باشد.

2 - printf("%s ", *--*++cpp+3);

در عبارت cpp++*--*+3 می آییم مرحله به مرحله بررسی میکنیم.ابتدا درcpp میدانیم که به آدرس خانه 1 از cp اشاره میکند چراکه در مرحله قبل cpp یک خانه جلو آمده بود. در مرحله بعدcp++ به آدرس خانه 2 از cp اشاره میکند. سپس cp++* به مقدار آن که c+1 است اشاره میکند که آن خود به آدرس خانه 1 از c اشاره میکند. پس cp++*-- به خانه ی قبل تر یعنی خانه 0 از c

اشاره میکند. حال cp++*--* ب مقدار آن خانه که عبارت "AUTPP" است را در بر دارد. در نتیجه cp++*--*3+ به خانه ی سوم این رشته میرود که کاراکتر 'P' می باشد و از آنجا به بعد را پرینت میکند. پس خروجی این printf عبارت "PP" می باشد.

3 - printf("%s ", *cpp[-2]+3);

پس از آنکه اشاره گر cpp بعد از مراحل 1 و 2 دوخانه به جلو آمد ودر آنجا ست شده حال cpp[-2] به دو خانه عقب تر اشاره میکند و اکنون به مقدار خونه 0 از cp که آدرس خونه 3 از c می باشد اشاره میکند.پس cp[-2]* مقدار آن خانه یعنی "QUIZ" می باشد. حال cpp[-2]*3+ به خانه سوم این رشته یعنی کاراکتر'Z' اشاره میکند و از آننجا به بعد را چاپ میکند و چون بع از آن حرف دیگری نیست پس خروجی اینprintf میشود: "Z"

4- printf("%s ", cpp[-1][-1]+1);

گفتیم که cpp پس از مراحل 1و 2 دو خانه جلو آمد و در آنجا ست شد. حال cpp[-1] به مقدار یه خانه عقب تر در cp اشاره میکند که خانه 1 از cp میباشد که آدرس خونه 2 از c در آنجاست. پس cpp[-1][-1] به مقدار خانه قبلی آن از c یعنی عبارت "principal" اشاره میکند. پس cpp[-1][-1]+1 به یک خانه جلو تر در این رشته یعنی کاراکتر 'r' اشاره میکند و از آنجا به بعد را در خروجی پرینت میکند. پس خروجی این printf میشود:"rincipal"

توجه شود که در هر printf یک space بعد ازs% میباشد پس خروجی اینprintf ها با یک فاصله از هم و در یک خط چاپ میشوند.

پ:

خروجی کد داده شده به شکل زیر است:

```
[2][0] = 5
[2][1] = 10
[2][2] = 14
[2][3] = 17
[2][4] = 19
[1][0] = 3
[1][1] = 6
[1][2] = 8
[0][0] = 1

--------------------------------
Process exited after 0.01873 seconds with return value 0
Press any key to continue . . .
```

توضیحات:
درواقع p2 و p3 را میتوان گفت هردو آرایه ای دو بعدی هستند که آدرس خانه هایشان یکی است. آرایه ای دو بعدی با 3 سطر و تعداد ستون های مختلف به این ترتیب که سطر اول یک ستون دارد و مقدار 1 در آن قرار دارد و در این حالت چونj=1 است پس وارد لوپ بعدی نمیشود.سطر دوم 3 ستون دارد که به ترتیب مقادیر 3و6 و8 میگیرد. به این ترتیب که خونه اول مقدار j را میگیرد و چون j>1 میباشد وارد لوپ بعدی شده وp1 که به اولین خانه این سطر اشاره میکرده یک واحد در آدرس جلو رفته و خانه دوم مقدار دهی میشود که برابر جمع j با مقدار خانه قبلی است. سپس j یک واحد کم شده و همین فرآیند تکرار میشود.

درنهایت به سطر سوم رفته که دارای 5 ستون است که به ترتیب دارای مقادیر 5 و 10 و 14 و 17و 19 هستند. روند مقدار دهی هم مانند سطر قبل میباشد و ستون اول برابر مقدارj است و سپس وارد لوپ بعدی شده و ستون های باقی مانده پر میشوند با همان روند ذکر شده.

در قسمت پرینت کردن هم به کمک p3 پیمایش را روی آرایه ساخته شده انجام داده به این صورت که اول از سطر سوم شروع کرده و از ستون اول تا پنجم آن را هرکدام در یک سطر و سپس سطر دوم از ستون اول تا سوم هرکدام در یک سطرو در آخر هم سطر اول آرایه و ستون اول آن نمایش داده میشود. شکل حافظه نیز به صورت زیر است: