گزارش کار HW2

پارسا اسدی 9523005

(Q1

کد 1)

این کدکامپایل می شود. ابتدا متغیری از نوع int به نام a تعریف شده است. سپس یک پوینتر از نوع *int تعریف شده است. از نوع تعریف پوینتر میتوان فهمیدکه خود این پوینتر constاست و نمیتوان ادرس جدیدی در آن نوشت. اما میتوان از طریق این پوینتر متغیر را که به آن اشاره میکند عوض کرد. در خط بعداز طریق پوینتر به متغیر b یک واحد اضافه می شود و در خط بعد مستقیما به خود a دوباره یک واحد اضافه می شود بنا براین با دستور چاپ در خط بعد مقدار متغیر a یعنی 4 چاپ میشود. همچنین پوینتر b که ادرس یک خانه در حافظه در آن ذخیره شده است(ادرس a) نیز چاپ میشود.

کد 2)

کدکامپایل نمی شود. ابتدا متغیری از نوع const int به نام a تعریف شده است.سپس دو متغیر z=از نوع intبه نام های c و d تعریف شده است.سپس یک پوینتر از نوع *int تعریف شده به نام b که به a اشاره میکند از نحوه تعریف این پوینتر می توان فهمیدکه خود این پوینتر const نیست ولی نمیتوان از طریق آن مقدار متغیری که به آن اشاره میکند را تغییر داد. در بخش بعدی ادرس c در پوینتر b نوشته می شود(این کار مشکلی ندارد زیرا خود پوینتر const نیست)از این به بعد این پوینتر به c اشاره می کند.در خط بعد اگر کدکامپایل می شدمتغیر aو پوینتر b و محتوای پوینترb چاپ میشد.(مقدار a 10 است و مقدار b ادرس خانه ای از حافظه است که c در آن ذخیره شده است و محتوای پوینتر همان c است که 20 است). در خط بعد یکپوینتر از نوع *int به نام e ایجاد می شود که آدرس c در آن ذخیره می شود. از نوع تعریف این پوینتر مشخص است که خود این پوینتر const است اما میتوان از طریق آن متغیری که به آن اشاره می کند را تغییر داد. در خط بعد برنامه نویس می خواهد ادرس dرا در e ذخیره کندکه error برنامه در همین خط است زیره همانطورکه گفتم خود پوینترconst است و نمیتوان ادرس جدیدی در آن نوشت. در خط بعد متغیر d و پوینترe و محتوای پوینتر e قرار است چاپ شود کهاگرکدکامپایل می شد چاپ میشد(مقدار d 30است و پوینترe ادرس خانه c در آن ذخیره شده و محتوای پوینتر e هم همان c است که مقدارش 20 است)

کد 3)

نوع متغیر خط 3 : متغیر از نوع *const charاست(به نام p1) که آدرس عنصر اول رشته ی کاراکتر name یعنی "A" در آن ذخیره شده (خانه های بعد از p1 در حافظه به ترتیب آدرس کاراکتر های بعدی رشته اند). از نوع تعریف پوینتر می توان فهمید خود پوینتر const نیست اما نمی توان از طریق آن رشته ای که به آن اشاره میکند را تغییر داد.

حکم خط 5 : مجاز است زیرا همانطورکه گفتم خود پوینترconstنیست و میتوان ادرس جدیدی در آن ذخیره کرد.

حکم خط 8 : مجاز نیست وerror دارد زیرا همانطورکه گفتم نمیتوان از طریق p1 رشته را تغییر داد.

حکم خط 9 : مجاز نیست و error دارد زیرا پوینتر از نوع *char برای اشاره به یک متغیر از نوع char است ولی رشته که میان double quotes است از نوع *const char است و یک پوینتر *char نمیتواند به آن اشاره کند

کد 4)

خط 1 : آرایه تک بعدی 10 تایی از نوع int که با new کردن پوینتر p1 به دست می آید

خط 2 : آرایه 10 تایی از نوع int* (آرایه ای از پوینتر ها)

خط 3 : function pointer به نام p3 که آرایه تک بعدی از نوع int را ورودی می گیرد.

خط 4 : آرایه 10 تایی از function pointer ها است که هرکدام از این function pointer ها آرایه دو بعدی که تعداد سطر هایشان مشخص نیست ولی 10 ستون دارند را به عنوان ورودی میگیرند.

خط 5 : آرایه دو بعدی 10*10 از نوع int . تفاوتش با آرایه دو بعدی معمولی این است که بر خلاف آرایه دو بعدی دینامیک معمولی هر 100 خانه پشت سر هم allocate میشوند .

کار اضافه ـ ـ درکدی که در پوشه Q1 است من آدرس تمام 100 خانه آرایه که به این ترتیب تعریف شده را به ترتیب چاپ کردم که مشاهده می شود تمام خانه ها به ترتیب اند. همچنین خانه های آرایه دو بعدی معمولی 10*10 را نیز چاپ کردم که شاهده می شود خانه های به فرم 10*n ام و 10*n+1 پشت سر هم نیستند!

Q2)

توابع مورد اسفاده:

الگوریتم به این شکل است که ابتدا کل فایل را در قال یک string میخوانیم. سپس با معرفی علایم نگارشی وبه طورکلی کاراکتر های که جزو حروف الفبا نیستند فایل را pars میکنیم و در یک vector تمام کلمات فایل را ذخیره میکنیم. میدانیم کلمات دو دسته اند :1)کلماتی که فقط از حروف الفبا تشکیل شده اند 2)کلماتی که ممکن است میان آن ها character های غیرالفبایی باشد مانند : space-X و SS_spring و غیره.ادامه الگوریتم به این شکل است که یک تابع به نام split_1 نوشتم که کلمات دسته دوم را نیز از به کلماتی میشکندکه خروجی های این تابع کلماتی از دسته اول اند. یک تابع split_2 نیز نوشتم که کلمات دسته اول را از محل کاراکتر های a,e,I,o,u,A,E,I,O,U برش میدهد. خروی این تابع کلماتی هستندکه فقط از حروف بی صدا تشکیل شده اند. حال باید این کلمات چک شوندکه 5 حرف صدادار یا بیشتر پشت سر هم در آن ها نیامده باشد. این کار با تابع check_normal_str انجام میپذیرد وخروجی یک Boolean است.

بخش اضافه بر سوال که انجام دادم: در سوال کلمات نوع دوم فقط به طوری در نظرگرفته میشود که کاراکتر "ـ" در میان آن باشد. اما کد من تمام کاراکتر های غیر الفبایی دیگرکه در میان کلمات امده اند را میفهمد.مثلا اگرکلمه همراه apostrophe باشد آن را تشخیص میدهد مانند I'am و it's و can't و ...

نحوه کار check_normal_str:

ورودی: string

خروجی: Boolean (خروجی true برای رشته های واجد شرایط صورت سوال و false برای بقیه ورودی ها)

چون ورودی این تابع کلمات دسته اول هستند پس ابتدا در این تابع آن ها را از صافی تابع split_2 میگذرانیم و خروجی آن زیر رشته هایی است که همگی از حروف بی صدا تشکیل شده اند.حال با حلقه ساده همه این زیر رشته ها را چک میکنیم که ببینیم ایا 5 حرف بی صدا در آن ها اماده است یا نه. اگر حداقل در یکی از زیر رشته ها 5 حرف بی صدا یا بیشتر پشت سر هم امده بود true بازگردانده میشود. در غیر این صورت falseبازگردانده میشود.

نحوه کار check_str :

ورودی: string

خروجی: Boolean (خروجی true برای رشته های واجد شرایط صورت سوال و false برای بقیه ورودی ها)

ورودی این تابع کلماتی هستندکه در آن ها کاراکتر های غیر الفبایی وجود دارند. پس باید ابتدا این ورودی ها را از صافی تابع split_1 بگذرانیم حال خروجی های این تابع همه رشته های از نوع اول هستند.پس کافیست آن ها را ب تابع check_normal_str بدهیم.

به عنوان جمع بندی نحوه کار برنامه به این صورت است :

1)ابتدافایل به کلمات سازنده اش تجزیه میشود و در vector ذخیره میشود.

2)سپس کلمات دسته اول تشخیص داده میشوند و به تابع check_normal_str تحویل داده میشوند.

3)سپس کلمات دسته دوم نیز به تابع check_str تحویل داده میشوند

4)کلماتی که به ازای آن ها خروجی دو تابع بالا برایشان true است چاپ میشوند

(Q3

کد جزییات زیادی دارد و ذکر همه ی آن ها درگزارش به اطناب بی موردی می انجامد بنابر این من الگوریتم و بخش های مختلف کلاس را به صورت کلی مطرح میکنم.

variableهای مورد نیازکلاس :

ارایه دینامیک که عناصر در آن ذخیره می شود(nums) و ارایه دینامیک از جنس *bool که هم اندازه با ارایه nums است(flags) (اگر خانه jام nums پر باشد خانه jام flags 1 است در غیر این صورت 0 است.) متغیر size از نوع int که اندازه ارایه دینامیک است. دو متغیر از نوع intبه نام های writer_pوreader_p که مکان pointerها را روی ارایه نشان میدهد. و در نهایت متغیر به نام capacity از نوع intکه مقدار جای خالی در صف را نشان می دهد.

Methodهای مهم کلاس:

1)constructor: فایل را به عنوان ورودی می گیرد. ارایه های دینامیک کلاس همینجا new میشوند. همچنین writer_pو pointer_p و capacity هم همینجا تعیین میشود.

بخش اضافه بر صورت سوال که انجام دادم: با catch و throw عمل error handlingرا انجام دادم که اگر فایل به هر دلیلی درست باز نشد errorبدهد.

2)enQueue: در محل فعلی writer_pمینویسد سپس writer_p را به محل جدیدش منتقل می کند اگر سمت راستش خالی بود و از ارایه بیرون نمی زد یکی به سمت راست میرود در غیر این صورت به ابتدای ارایه یعنی خانه صفر می رود.(البته در ابتدا چک میشود که ارایه پر نباشد)

3)dequeue: خانه که reader_p روی آن است خالی میکند و سپس reader_p را به محل جدیدش منتقل می کند. اگر سمت راستش خالی بود و از ارایه بیرون نمیزد یکی به سمت راست میرود در غیر این صورت به ابتدای ارایه یعنی خانه صفر می رود.(البته در ابتدا چک میشود که ارایه خالی نباشد)

4)displayQueue: به این صورت است که از خانه reader_p به سمت راست چاپ میکند تا انتهای ارایه. سپس از ابتدای آرایه تا reader_p چاپ میکند به این ترتیب ما صف را به درستی مشاهده میکنیم.

5)destructor: تمامتغیر های دینامیک کلاس را پاک میکند.

(Q4

توابع مورد استفاده (سوال بسیار ساده است بنابراین فقط توابع مورد استفاده را معرفی میکنم)

Find_min: یک ارایه از اعداد را به عنوان ورودی میگیرد وکوچکترین عنصر را به عنوان خروجی میگیرد.

Replace: دو int به نام های aو b میگیرد و یک ارایه نیز میگیرد . و جای دو عنصر a ام و b ام ارایه را عوض میکند.

Sort: با استفاده از دو تابع بالا الگوریتم selection sort را اجرا می کند