

محمد صادق قصری زاده دزفولی

گزارش تمرین سری سوم

درس: برنامه نویسی پیشرفته

دکتر جهانشاهی

آدرس در GitHub :

سوال اول از بخش فرانت :

در این سوال هدف ساختن بازی نقطه و خط بود به نحوی که آنرا با زبان های تحت وب بتوان نوشت و اینکه این نمونه سایت می بایست ریسپانسیو می بود؛ برای ریسپانسیو کردن این نمونه سایت آمدم برای ارتفاع دیو هایی که ساختم به جای واحد px از واحد vh استفاده کردیم تا با تغییر ارتفاع صفحه ، ارتفاع دیو ها نیز تغییر کند. همچنین برای عرض (width) دیو ها از واحد vw استفاده کردیم تا با تغییر عرض صفحه نیز عرض دیو ها تغییر کند. خب برای ساخت بازی ابتدا اینجانب هر ردیف از نقاط و خطوط عمودی و خطوط افقی را یک دیو در نظر گرفتم همچنین هر خط عمودی را در یک دیو که زیر مجموعه دیو نقطه بالای آن است در نظر گرفتم. بین نقاط نیز از دیو هایی برای تشکیل خطوط افقی استفاده کردم.

خب در صفحه ی اولیه بازی، اطلاعاتی نظیر اسم بازیکنان (چه قرمز و چه آبی) و همچنین تعداد نقاط افقی و عمودی را دریافت میکنیم تا به وسیله جاوا اسکریپت صفحه بازی را با آن تعداد نقاط وارد شده تشکیل دهیم؛ که کد آنرا به همراه تمارین دیگر ارسال نموده ام.

خب وارد صفحه بازی شدیم!.. در بالای صفحه بخشی برای نمایش امتیاز هر بازیکن وجود که نمایش دهنده تعداد مربع هایی است که آن بازیکن تشکیل داده است. همچنین یک بخش برای نمایش دادن اینکه در این مرحله نوبت کدام بازیکن برای انجام بازی است وجود دارد.

**** بعد از تشکیل مربع به وسیله یک بازیکن ، نفر بعدی که بازی میکند همان بازیکن است.**

خب برای تشکیل خطوط بین نقاط و رنگی کردن آن ها از کد جاوا اسکریپت استفاده کرده و به وسیله کلیک روی آن دیو، آن دیو رنگی میشود؛ به رنگ بازیکنی که نوبتش بود!...

***همچنین برای راحتی کار بازیکن، روی هر خط (چه افقی و چه عمودی) یک hover تعریف کرده ام که به محض اینکه بازیکن ماوس را روی آن خط برد (قبل از اینکه کلیک کند) رنگ آن از سفید به خاکستری تغییر میکند که به محض کلیک روی آن خط ، رنگی میشود...**

همچنین بعد از اینکه تمام خطوط رنگی شدند و تمام مربع ها پر شد، به معنای این است که بازی تمام شده است. در این حالت پس از رنگی شدن آخرین خط و تشکیل آخرین مربع، صفحه ی جدیدی باز میشود و برنده را اعلام میکند و از صفحه بازی خارج میشویم. همچنین اگر امتیاز هر دو بازیکن با هم برابر بود بازی مساوی اعلام میکند.

Turn : Blue

نمای صفحه بازی:

Red Score : 0

Blue Score : 0



سوال اول از بخش Cpp :

در این سوال هدف ساختن یک ساختمان مانند، به نام maxheap است که الگوریتم ساخت آن به این صورت است که گره ی والد باید از تمام گره های ولد بزرگتر باشد . هرگره ی والد یک ولد سمت راست و یک ولد سمت چپ دارد که هر دوکو چکترند از آن گره. پس در بالاترین گره این ساختار، بزرگترین مقدار تمام ساختار وجود دارد. به عبارتی المان شماره ((۱)) این ساختار ماکزیمم تمام مقادیر موجود در این ساختار است.

لذا یک کلاس به نام maxheap میسازیم و توابع توضیح داده شده در صورت سوال را در آن تعریف میکنیم.

مثلا توابعی که در main صدا زده شده اند مانند , add(),parent() , leftChild(),rightChild(),heapsort(),printArray(),...

توابع نامبرده شده و تمام توابع موجود در داخل کلاس به درستی کال شده اند و عملکرد درستی دارند.

همچنین اپراتور های خواسته شده در صورت سوال نیز به طور کامل نوشته اند.

نمونه خروجی این برنامه را مشاهده میکنیم:

```
saeghgasr@saeghgasr:~/HW3/Cpp/Q1$ ./main
101 ,
32 , 17 ,
23 , 25

101
32
17

75 ,
52 , 23 ,
1 , 11 , 10 , 7

height of h2 is 2
75,52,23,11,10,7,1,
75
52
50
saeghgasr@saeghgasr:~/HW3/Cpp/Q1$
```

همانطور که میبینیم تمام خروجی مطابق آنچه است که در main به صورت کامنت نوشته شده اند

سوال دوم از بخش Cpp :

در این سوال هدف این است که کلاسی مانند وکتور تعریف کنیم و بتوانیم از آن آبجکت بسازیم؛ که البته در ساخت این این کلاس از آرایه های دینامیک استفاده شده است تا سرعت کار را بالاتر ببرد. همچنین آبجکت های این کلاس قابلیت `pushback` کردن نیز دارد به این صورت که با هر بار اجرای این تابع آرایه ی قبلی پاک شده و اطلاعات آنرا در یک آرایه جدید که آنرا `new` کرده ایم انتقال میدهم و طول آن نیز یکی اضافه میشود. همچنین یک آبجکت از کلاس وکتور دارای خاصیت `capacity` میباشد. در بخش دیگری از کلاس اپراتور های ضرب نقطه ای جمع داخلی نیز برای این کلاس تعریف شده است که المان به المان می آید و در هم ضرب و یا با هم جمع می کند. همچنین اپراتور های `<` و `>` و `=` را برای این کلاس نیز تعریف می کنیم؛ به طوری که بر اساس سائز آن وکتور ها مقایسه را انجام می دهد.

در این سوال طبق صورت سوال فرض شده است که تمام المان های وکتور از نوع `int` می باشد... که اگر بخواهیم هر نوع تایی را بگیرد به کلاس تمپلیت نیاز است.

**** برای کانستراکتور این کلاس از `initializer_list` استفاده شده است تا بتوان با دادن یک لیست از داده ها نیز بتوان یک آبجکت از کلاس را ساخت.**

همچنین دو نوع کانستراکتور برای کلاس داریم یکی از نوع `Copy` و یکی از نوع `Move` میباشد که تفاوت این دو نوع کانستراکتور این است که در کانستراکتور نوع `Copy` می آید و المان به المان، المان های درون آرایه ی آبجکت اول را به المان های درون آرایه ی آبجکت دوم کپی کرده و سپس آرایه ی آبجکت اول را حذف میکند؛ ولی در کانستراکتور از نوع `Move` می آید و نام آرایه ی آبجکت دوم را با نام آرایه ی آبجکت اول برابر قرار داده که یعنی بتوان از آرایه ی آبجکت دوم به المان های آرایه ی آبجکت اول دسترسی داشته باشیم و سپس آرایه ی آبجکت اول را برابر با `nullptr` قرار میدهم... این تفاوت دون کانستراکتور است.

*البته بدی کانستراکتور از نوع `Move` این است که بعد از کال شدن این کانستراکتور دیگر به آرایه ی آبجکت اول دسترسی نداریم.

```
saeghgasr@saeghgasr:~/HW3/Cpp/Q2$ make
g++ -std=c++17 -Wall -c main.cpp
g++ -std=c++17 -Wall -c -c -o vector.o vector.cpp
g++ -std=c++17 main.o vector.o -o main
saeghgasr@saeghgasr:~/HW3/Cpp/Q2$ ./main
move constructor
10
10 ,9 ,8 ,7 ,6 ,5 ,4 ,3 ,2 ,1 ,
1 ,2 ,3 ,4 ,5 ,6 ,7 ,8 ,9 ,10 ,
saeghgasr@saeghgasr:~/HW3/Cpp/Q2$
```

خروجی را میبینیم:

```
1 #include <iostream>
2 #include "vector.h"
3
4
5 int main()
6 {
7     Vector v{1,2,3,4,5,6,7,8,9,10};
8     Vector v1{10,9,8,7,6,5,4,3,2,1};
9     Vector b{std::move(v1)};
10    //while(v1 < v)
11    //    v1.push_back(1);
12    std::cout <<v.size<<std::endl;
13    //    v1.show();
14    b.show();
15
16
17    return 0;
18 }
```

Main این برنامه به صورت زیر است:

سوال سوم از بخش Cpp :

در این سوال هدف سازمان بندی و طبقه بندی افراد درون یک شهر است به طوری که هیچکدامشان هویتشان را یادشان نمی آید. در این سوال، افراد را که با نام human معرفی کرده ایم به صورت پویتری ذخیره میکنیم به شکل *human که این خاصیت بعد ها به کمک ما می آید تا بتوان راحت تر دسته بندی افراد را انجام دهیم. هر آبجکت (فرد) که از کلاس human ساخته میشود دارای هفت صفت (نام، نام خانوادگی، رنگ مو، رنگ چشم، جنسیت، تعداد فرزندان و سن) را دارد که تمام آن ها باید در کانستراکتور تابع مقداردهی بشوند.

توابع موجود را ببینیم:

اپراتور <: که برای مقایسه سن دو فرد است که به صورت بولین تعریف میشود.

اپراتور +: که برای این است که در صورتی دو نفر همسر یکدیگر باشند بتوانند یک human را بسازند و آن فرزندان است.

اپراتور ++: که سن فرد را یکی زیاد میکند

و دیگر توابعی که در صورت سوال آورده شده است.

همچنین این کلاس به کپی کانستراکتور نیاز دارد.

چون طبق صورت سوال باید صفت های فرد از کلاس human به صورت private باشند لذا باید توابعی را برای نمایش دادن آن ها و همچنین مقدار دادن به آن ها تعریف نمود که از این طریق به آنها بتوان دسترسی یافت.

خب برای تنظیم کردن و دسته بندی اطلاعات به کلاسی دیگر به نام

Oracle نیاز داریم که به وسیله توابع موجود در آن میتوان دو نفر را که قبلا با هم هیچگونه رابطه ای نداشته اند را به ازدواج هم درآورد و خانواده را تشکیل داد و سپس به وسیله آن بتوان از تعدادی از اپراتور کلاس human استفاده کرد..

در این کلاس نیز تابعی برای set کردن فرزند وجود دارد که یک فرد را فرزند دو فرد دیگر قرار میدهد.

**هر آبجکت از کلاس human دارای سه *human به نام های

Father, mother, spouse دارد که به ترتیب نماینده پویتری از پدر، مادر و همسر است. نمونه خروجی را به صورت میبینیم:

```
saeghgasr@saeghgasr:~/HW3/Cpp/Q3$ ./main
BAHAR
girl
saeghgasr@saeghgasr:~/HW3/Cpp/Q3$
```

که main آن به صورت رو به رو است:

```

1  #include <iostream>
2  #include "human.h"
3  #include "oracle.h"
4
5  int main() {
6
7
8
9      human* p1{new human {"ALI", "BAHADORI", 4, 3, 25, 0, 0}};
10     human* p2{new human {"BAHAR", "SHAMS", 1, 2, 22, 1, 0}};
11     human* p3{};
12
13     Oracle o1{"SHERVIN"};
14     o1.marry(p1, p2);
15     std::cout << (*(p1->spouse)).getFirstName()<< std::endl;
16     //married couple
17
18     // std::cout << (*(p2->spouse)).getEyeColor() << std::endl;
19     //creating new person*/
20     p3 = (*p1) + (*p2);
21     if(p3->getGender())
22     {
23         p3->setFirstName("MAHSHID");
24         std::cout<< "girl"<<std::endl;
25     }
26     else
27     {
28         p3->setFirstName("FARSHID");
29         std::cout << "boy"<<std::endl;
30     }
31
32     p3++;
33
34     human* p4{new human {"SHAHIN", "REZAEI", 0, 5, 58, 1, 1}};
35     human* p5{new human {"FARHAD", "BAHADORI", 0, 6, 63, 0, 1}};
36
37
38     o1.setChild(p1, p4, p5);

```

سوال اول از بخش DataBase :

در این سوال هدف نوشتن یک پایگاه داده مانند پایگاه داده تلگرام است.

در این پایگاه داده به یک سری جداول نیاز داریم که در صورت سوال داده شده است.

این جداول را میسازیم و سپس از طریق دستور insert به آن جدول ها ردیف اضافه میکنیم.

دستور کوئری ها را به صورت زیر میبینیم.

سوال ۱ :

id	telegram_id	phone	email	first_name	last_name	verification_code	profile_picture_url	created_at	updated_at	password
2	@sadeh	0912000001	sadeh@gmail.com	sadeh	ghasrizadeh	fisrt	desktop	2019-04-10 21:53:10.301216	2019-04-10 21:53:10.301216	395a625935d2ce068d549b05711df4c

سوال ۲ :

```
000001';=# UPDATE Users SET email = 'apstudent2019@gmail.com' WHERE phone='091200
UPDATE 1
postgres=#
```

سوال ۳ :

```
000001';=# UPDATE Users SET email = 'apstudent2019@gmail.com' WHERE phone='091200
UPDATE 1
M Users WHERE phone = '0912000001');nnelmembership where user_id=(SELECT id FROM
channel_id
-----
@aut ap 2019
@gizmiz
(2 rows)
postgres=#
```

سوال ۴ :

```
19'; res=# SELECT COUNT(*) FROM channelmembership WHERE channel_id = '@aut_ap_201
count
-----
5
(1 row)
postgres=#
```

سوال ۵ :

```
postgres=# SELECT email FROM Users WHERE phone LIKE '0935_%';
email
-----
hassan@gmail.com
taghi@gmail.com
amir@gmail.com
shd@gmail.com
apstudent2019@gmail.com
(5 rows)
```

سوال ۶:

```
postgres=# SELECT phone FROM Users WHERE telegram_id in(SELECT blocked_user_id FROM BlockUser WHERE blocker_user_id = (SELECT telegram_id FROM users WHERE phone = '09120000001') AND (ecrated_at BETWEEN now() - INTERVAL '1' month AND now()));
           phone
-----
091200000054
09357654321
(2 rows)
```

سوال ۷:

```
postgres=# SELECT message_text FROM message WHERE (sender_id = '@amir.jahanshahi' AND reciever_id = '@sHdIV4RHs') OR (reciever_id = '@amir.jahanshahi' AND sender_id = '@sHdIV4RHs') ORDER BY id DESC LIMIT 10;
           message_text
-----
I Have DB Exercise... I will answer you tomorrow!!! GoodBy :)
now7
now 6
now 5
:)
BYE
now 4
now3
now2
now1
(10 rows)

postgres=#
```

سوال ۸:

```
postgres=# INSERT INTO BlockUser(blocker_user_id,blocked_user_id,ecrated_at)
postgres=# VALUES((select telegram_id from users where phone='09120000001'),(select telegram_id from users where telegram_id='@taghi'),now());
ERROR:  duplicate key value violates unique constraint "blockuser_pkey"
DETAIL:  Key (blocker_user_id, blocked_user_id)=(@sadegh, @taghi) already exists.
postgres=# select* from Blockuser;
 blocker_user_id | blocked_user_id | ecrated_at
-----
@sadegh          | @ali            | 2019-04-11 11:34:55.942879
@sadegh          | @taghi          | 2019-04-11 11:34:55.994037
(2 rows)

postgres=# SELECT phone FROM Users WHERE telegram_id in(SELECT blocked_user_id FROM BlockUser WHERE blocker_user_id = (SELECT telegram_id FROM users WHERE phone = '09120000001') AND (ecrated_at BETWEEN now() - INTERVAL '1' month AND now()));
           phone
-----
091200000054
09357654321
(2 rows)

postgres=# SELECT message_text FROM message WHERE (sender_id = '@amir.jahanshahi' AND reciever_id = '@sHdIV4RHs') OR (reciever_id = '@amir.jahanshahi' AND sender_id = '@sHdIV4RHs') ORDER BY id DESC LIMIT 10;
           message_text
-----
I Have DB Exercise... I will answer you tomorrow!!! GoodBy :)
now7
now 6
now 5
:)
BYE
now 4
now3
now2
now1
(10 rows)

postgres=# SELECT phone FROM Users WHERE telegram_id in (SELECT creator_id FROM channel WHERE updated_at NOT BETWEEN now() - INTERVAL '1' month AND now() );
           phone
-----
091200000001
(1 row)

postgres=# select* from message;
V4RHs') ORDER BY id; OR (reciever_id = '@amir.jahanshahi' AND sender_id = '@sHdIV
ERROR:  syntax error at or near "ORDER"
LINE 1: ... '@amir.jahanshahi' AND sender_id = '@sHdIV4RHs') ORDER BY i...
V4RHs');@sHdIV4RHs') OR (reciever_id = '@amir.jahanshahi' AND sender_id = '@sHdIV
DELETE IS
postgres=# select* from message;
 id | sender_id | reciever_id | message_type | message_text | created_at | updated_at
-----
 3 | @ali     | @sadegh    | text         | Hello How are you? | 2019-04-11 16:40:19.531169 | 2019-04-11 16:40:19.531169
 4 | @sadegh | @ali       | text         | Hello How are you? | 2019-04-11 16:40:19.532474 | 2019-04-11 16:40:19.532474
(2 rows)

postgres=#
```

سوال ۹:

```
postgres=# SELECT phone FROM Users WHERE telegram_id in (SELECT creator_id FROM channel WHERE updated_at NOT BETWEEN now() - INTERVAL '1' month AND now() );
           phone
-----
091200000001
(1 row)
```


جواب سوال پرسیده شده:

برای ادمین معرفی ادمین ها میتوان آرایه تعریف کرد به صورت `int[]`
تعریف میکنیم.