

بسمه تعالی

## تمرین دوم ساختمان داده ها

مهلت انجام : برای گروه یکشنبه، ۴ آبان ماه و برای گروه سه شنبه ۶ آبان ماه

۱. تمام تمرینات به صورت انفرادی انجام خواهند گرفت.
۲. نام و نام خانوادگی ، شماره دانشجویی و شماره تمرین در بالای صفحه نوشته شود.
۳. پاسخ تمرین خود را به صورت دستی در کلاس تحویل دهید و از ارسال آن از طریق ایمیل خودداری کنید.
۴. تمرینات به صورت دستی نوشته شود و از تایپ آن خودداری شود.

۱- تابع زیر به صورت بازگشتی متقارن بودن یک رشته را بررسی می کند. پیچیدگی زمانی این تابع را به دست آورید.

```
1 int isPalindrome ( char A[] , int n ){
2     if (n <= 1) return 1;
3     if (A[0] != A[n-1]) return 0;
4     return isPalindrome(&A[1],n-2);
5 }
```

۲- پیچیدگی زمانی مسئله ی برج هانوی را همراه با توضیح کامل به صورت یک رابطه ی بازگشتی بنویسید، سپس مرتبه ی زمانی رابطه ی بازگشتی به دست آمده را با استفاده از روش درخت حدس بزنید؟

۳- فرض کنید برنامه ای با اندازه ورودی ۶۰۰، به مدت ۳۰ ثانیه و با اندازه ورودی ۱۸۰۰، به مدت ۴,۵ دقیقه اجرا می شود. پیچیدگی زمانی این برنامه از چه تابعی است؟

۴- با در نظر گرفتن شبه کد، به سوالات زیر پاسخ دهید:

الف) زمان اجرا را برای این برنامه را به دست آورید؟

ب) با تغییر دادن خط هفتم برنامه به  $Z = 1$  و خط چهاردهم برنامه به  $Z = Z + Z$  زمان اجرای برنامه را به دست آورید؟

```

1 int mysteryAlgorithm(n)
2 local variables
3     x, y, z: INTEGER
4 begin
5     x = 0
6     y = 1
7     z = 0
8     while y <= n do
9         if x = y then
10             y = y + 1
11             x = 0
12         endIf
13         x = x + 1
14         z = z + 1
15     endwhile
16     returnvalue = z
17 end

```

۵- دو تابع زیر را در نظر بگیرید:

الف) با فراخوانی  $F1(4)$  مقادیری را که در خروجی چاپ می‌شود، به دست آورید (ترتیب فراخوانی‌های انجام شده را ذکر کنید).

ب) با فراخوانی  $F2(4)$  مقادیری را که در خروجی چاپ می‌شود، به دست آورید (ترتیب فراخوانی‌های انجام شده را ذکر کنید).

```

1 void F1 (int x)
2 {
3     if (x>0)
4         F2(x-1);
5     printf(x);
6 }
7
8 void F2 (int y)
9 {
10    if (y>0)
11    {
12        printf(y+1);
13        F1(y-1);
14    }
15 }

```

۶- آرایه  $M[6...53][-15...16][20...55]$  را در نظر بگیرید. این آرایه از خانه‌ی ۵۰ حافظه ذخیره شده است و هر عنصر آرایه ۴ بایت اشغال می‌کند. آدرس خانه‌ی  $M[10][0][20]$  را به دست آورید در صورتی که:

الف) ماتریس به صورت سطری ذخیره شده باشد.

ب) ماتریس به صورت ستونی ذخیره شده باشد.

۷- آرایه  $A['a' \dots 'z'][-5...15][10...40]$  به صورت سطری در آرایه‌ی  $M[-10...8726]$  ذخیره می‌کنیم. عنصر موجود در  $A['e'][2][20]$  در کدام خانه از آرایه‌ی  $M$  قرار خواهد گرفت؟

۸- آرایه  $A[1..n]$  یک آرایه unimodal گوییم در صورتی که شرط زیر را رعایت کند:

$$A[i] < A[i+1] : 1 \leq i < m$$

$$A[i] > A[i+1] : m < i \leq n$$

در این صورت  $A[m]$  درایه ماکزیمم آرایه است. الگوریتمی برای تعیین ماکزیمم آرایه ارائه کنید که از مرتبه زمانی  $O(\log n)$  باشد.

۹- یک پشته خالی با اعداد ۰ تا ۹ در ورودی داده شده اند. اعمال POP و PUSH بر روی این پشته قابل انجام است. کدام یک از گزینه‌های زیر را نمیتوان با هیچ ترتیبی از اعمال فوق به دست آورد؟

الف) ۴۳۲۱۰۹۸۷۶۵

ب) ۴۶۸۷۵۳۲۹۰۱

ج) ۲۵۶۷۴۸۹۳۱۰

د) ۴۳۲۱۰۵۶۷۸۹

۱۰- زیربرنامه‌های زیر چه عملی را بر روی لیست پیوندی یک طرفه انجام می‌دهند؟ (همراه با توضیح).

<pre> 1  Void g(Node start){ 2      Node p,q,r; 3      P=start; 4      Q=null; 5      While(p != null){ 6          r=q; 7          q=p; 8          p=p.link; 9          q.link=r; 10     } 11     start=q; 12 }</pre>	<pre> 1  Node x(Node l){ 2      Node m,t; 3      m=null; 4      while(l != null){ 5          t=m; 6          m=l; 7          l=l.link; 8          m.link=t; 9      } 10     return m; 11 }</pre>
زیربرنامه اول	زیربرنامه دوم

۱۱- ساختمان داده ای بنویسید که با استفاده از دو پشته عملکرد یک صف را ایجاد کند؟

۱۲- اگر هر کاراکتر به معنی PUSH و هر ستاره (\*) به معنی POP باشد، خروجی حاصل از عمل POP را با در نظر گرفتن دنباله کاراکترهای زیر (از چپ به راست) بنویسید؟

E A S \* Y \* Q U E \* \* \* S T \* \* \* I O \* N \* \* \*

۱۳- تابعی بنویسید که iامین عنصر از لیست پیوندی یک طرفه را حذف کند (مطمئن شوید که iامین عنصر وجود دارد).

۱۴- تابعی بنویسید که مشابه بودن دو لیست پیوندی را بررسی کند.

۱۵- تابع زیر چه عملی را بر روی لیست پیوندی انجام می دهد؟

```

public void modify(SLList list) {
    if (list.head != null && list.head.next != null) {
        Node tmp = list.head.next;
        list.head.next = tmp.next;
        tmp.next = list.head;
        list.head = tmp;
    }
}
```

۱۶-تابع زیر چه عملی را بر روی لیست پیوندی انجام می‌دهد؟

```
public void modify(SLList list, int el) {  
    if (! list.isEmpty()) {  
        list.tail.next = new Node(list.tail.info);  
        list.tail.info = el;  
        list.tail = list.tail.next;  
    }  
}
```