

مدت آزمون: 150 دقیقه
دی 1389

به نام خدا
آزمون پایان ترم درس ساختمان داده ها

نام:
شماره دانشجویی:

- 1-** در هر یک از موارد زیر پیمایش خواسته شده را بدست آورید.
(الف) پیمایش Preorder یک درخت جستجوی دودویی BST با Postorder: 5, 6, 15, 10, 23, 24, 22, 26, 20 پیمایش Postorder به صورت زیر:
- Preorder
-
- (ب)** پیمایش Postorder یک درخت دودویی با پیمایشهای Inorder و Preorder: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12 به صورت روبرو:
- Inorder: 3, 5, 4, 2, 1, 9, 8, 10, 7, 11, 6, 12
- Postorder
-
- (پ)** پیمایش Levelorder یک درخت دودویی کامل (Complete) با Inorder: 60, 18, 61, 73, 22, 85, 75, 82, 45, 31, 6, 81, 16, 37 پیمایش Inorder به صورت روبرو:
- Levelorder
-
- 2- الف)** به یک Min-heap خالی گره هایی با کلیدهای (به ترتیب از راست به چپ) 68, 73, 18, 82, 45, 31, 6, 81, 16, 37, 22, 58, 12 اضافه شده است. Min-heap حاصل را رسم کنید.
- (ب)** پس از انجام سه عمل حذف (Delete) بر روی Min-heap بند الف، Min-heap حاصل را رسم کنید.
-
- 3-** یک لیست 12 عنصری حاوی کلیدهای 1 تا 12 به صورت مرتب صعودی است. اگر این لیست به صورت درجا تبدیل به یک MaxHeap شود، لیست نتیجه را نشان دهید.
-
- 4-** جدول درهم سازی HT[0..6] و تابع درهم سازی H(x) به صورت زیر را در نظر بگیرید.
 $H(x) = x \bmod 7$
اگر از روش آدرس دهی باز خطی برای مقابله با سرریز استفاده شود:
جدول درهم سازی را پس از درج عناصر (از چپ به راست) روبرو نشان دهید.
>> 12, 18, 27, 12, 5, 7, 13
-
- 5-** مرتبه زمانی هر یک از شبه کدهای زیر را بدست آورید.
- | | | |
|--|--|---|
| <pre>for (i=1; i<=n; i++) for (j=1; j<=n; j=j+i) x++;</pre> <p>(ج)</p> | <pre>for(i=1; i<=n; i++) { j=1; while(j<=i) { x++; j=j*2; } }</pre> <p>(ب)</p> | <pre>for(i=1; i<=n; i++) for(j=1; j<=n; j++) { x++; n--; }</pre> <p>(الف)</p> |
|--|--|---|
-
- 6- الف)** عبارت prefix زیر را به postfix تبدیل کنید:
 $++a/b-cd/-ab-+c*de/a-bc$

مدت آزمون: 150 دقیقه
دی 1389

به نام خدا
آزمون پایان ترم درس ساختمان داده ها

نام:
شماره دانشجویی:

ب) عبارت infix زیر را به prefix تبدیل کنید

$$a * b^{(c + d^{e * f}) / g - h * k}$$

7- مرتبه زمانی توابع بازگشتی زیر را بیابید:

$$T(n) = T\left(\frac{n}{3}\right) + T\left(\frac{2n}{3}\right) + n$$

8- نمایش preorder یک درخت دودویی به صورت XYZABVFOD است. این درخت گره تک فرزندی ندارد. همچنین گره های DOVBZ مجموعه برگهای درخت را تشکیل می دهند. درخت را ترسیم کنید.

9- الف) با n عدد متمایز چند درخت جستجوی دودویی BST می توان ساخت؟

ب) با همین تعداد داده متفاوت چند درخت دودویی می توان ایجاد کرد؟

10- الف) چند Max Heap با سه عنصر که حاوی کلید های 1 تا 3 (با هر ترتیب دلخواه) است، می توان ساخت.

ب) چند Max Heap با هفت عنصر که حاوی کلید های 1 تا 7 (با هر ترتیب دلخواه) است، می توان ساخت.

11- الف) حداکثر تعداد گره ها در یک درخت دودویی با ارتفاع h چقدر است؟

ب) حداکثر تعداد گره های برگ در یک درخت دودویی با ارتفاع h چقدر است؟

پ) حداکثر تعداد گره های با دوفرزند در یک درخت دودویی با ارتفاع h چقدر است؟

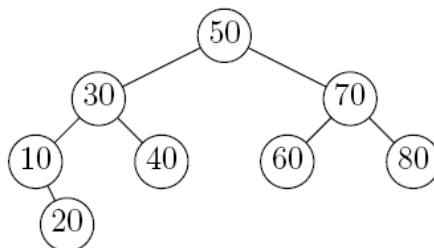
مدت آزمون: 150 دقیقه
دی 1389

به نام خدا
آزمون پایان ترم درس ساختمان داده ها

نام:
شماره دانشجویی:

(ت) عمق گره با اندیس i در یک heap با n گره چیست؟

(ث) با n گره، چند درخت دودویی متفاوت (از لحاظ توپولوژی) به ارتفاع n می توان ساخت؟



شکل 1 - درخت AVL

(ج) در درخت AVL فوق (شکل 1) اگر کلید 25 اضافه شود، Preorder درخت را نشان دهید.

(چ) در درخت AVL شکل 1 اگر کلید 65 اضافه شود، Preorder درخت را نشان دهید.

12- تابعی زیر چک می کند که آیا یک درخت دودویی با عناصر صحیح (int) و متمایز و با ریشه root یک درخت جستجوی دودویی است یا خیر. محلهای خالی A تا D در تابع زیر را کامل کنید.

```

bool IsBST(tree* t)
{
    Return IsBST(t, MININT, MAXINT);
}

int IsBST(tree *t, int min, int max)
{
    If (t==NULL) return TRUE;
    If (t->data<min || t->data>max) return FALSE;
    Return IsBST(t->LeftChild, ...A..., ...B...) && IsBST(t->RightChild, ...C..., ...D...);
}

```

13- یک گراف با 13 گره و 12 یال اگر بدون سیکل باشد، از چند مولفه به هم پیوسته (Connected Component) تشکیل شده است؟

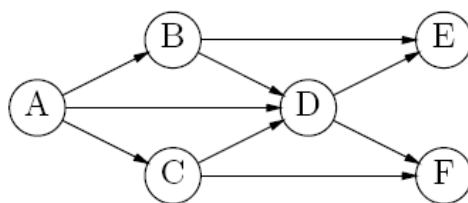
الف) 1 ب) 2 ج) 3 د) 4

گراف 1 را به شکل زیر در نظر بگیرید:

مدت آزمون: 150 دقیقه
دی 1389

به نام خدا
آزمون پایان ترم درس ساختمان داده ها

نام:
شماره دانشجویی:



گراف 1

14- کدام گزینه یک پیمایش درست جستجوی عمق اول DFS (Depth-First-Search) گراف شکل 1 نیست؟
 الف) ADEFBC (ب) ABEDFC (ج) ABDFEC (د) ABCDEF

15- کدام گزینه یک پیمایش درست جستجوی سطح اول DFS (Breadth-First-Search) گراف شکل 1 نیست؟
 الف) ABCDEF (ب) ABCDFE (ج) ADBCEF (د) ADBCFE

16- تابعی بنویسید که Max Heap با n عنصر A را به همراه اندیس i ($1 \leq i \leq n$) دریافت نماید و عنصر $A[i]$ را از Heap حذف کند به گونه ای که ساختار Heap آسیب نبیند و مرتبه زمانی الگوریتم $\log n$ باشد.

17- درخت جستجوی دودویی به صورت زیر تعریف شده است.

```

Class BSTNode
{
    BSTNode * LeftChild;
    int data;
    BSTNode * RightChild;
};
Class BST
{
    Public:
  
```

مدت آزمون: 150 دقیقه
دی 1389

به نام خدا
آزمون پایان ترم درس ساختمان داده ها

نام:
شماره دانشجویی:

Private:
BSTNode * root;
};

تابعی بنویسید که این درخت دودویی را تبدیل به یک لیست یک طرفه با کلیدهای مرتب صعودی و با آدرس شروع root کند. (به عبارتی اگر در آینده تابع چاپ از ریشه شروع به چاپ کند و از طریق فیلد RightChild پیش رود تا به null برسد، کلیدها به ترتیب صعودی چاپ شوند. در ضمن تنها از همین گره های درخت استفاده شود و با تغییر اشاره گره های گره ها تبدیل انجام شود).

14- تابعی بنویسید که یک گراف با n گره را از روی نمایش لیست مجاورت به ماتریس مجاورت ذخیره کند. ساختار کلاس های لازم را نیز نشان دهید.

مدت آزمون: 150 دقیقه
دی 1389

به نام خدا
آزمون پایان ترم درس ساختمان داده ها

نام:
شماره دانشجویی:

15- بر اساس تعریف کلاس لیست تعمیم یافته روبرو به سوال زیر پاسخ دهید.
تابع ~Genlist() را کامل کنید. این تابع لیست و ملحقات آن را حذف می کند.

```
class GenListNode
{ friend class GenList;
  Private:
    GenListNode *link;
    int tag; // 0 for data, 1 for dlink, 2 for ref
    union
    { int data;
      GenListNode *dlink;
      int ref;
    };
};

Class GenList
{ Private:
  GenListNode *first;
Public:
  GenList();
  ~GenList();
  void Print ();
  ...
};
```

مدت آزمون: 150 دقیقه
دی 1389

به نام خدا
آزمون پایان ترم درس ساختمان داده ها

نام:
شماره دانشجویی:

16- با توجه به کد Radix sort که در زیر آمده است. به سوالات زیر پاسخ دهید:

```
void RadixSort(Element *list, const int d, const int n)
{
    int e[radix], f[radix]; // queue pointers
    for (int i = 1; i <= n; i++) list[i].link = i+1;
    list[n].link = 0; int current = 1;
    for (i = d-1; i >= 0; i--)
    {
        for (int j = 0; j < radix; j++) f[j] = 0;
        for (;current; ..... A ..... ) {
            int k = list[current].key[i];
            if (f[k] == 0) ..... B .....;
            else list[e[k]].link = current;
            e[k] = current;
        }

        for (j = 0; f[j] == 0; j++);
        current = f[j]; int last = e[j];

        for (int k = j+1; k < radix; k++)
            if (f[k]) {
                ..... C .....;
                last = e[k];
            }
        ..... D .....;
    }
    // insert printing code
}
```

الف) در کد فوق جاهای خالی را تکمیل کنید.

مدت آزمون: 150 دقیقه
دی 1389

به نام خدا
آزمون پایان ترم درس ساختمان داده ها

نام:
شماره دانشجویی:

A=	C=
B=	D=

فرض کنید که لیست زیر ورودی الگوریتم باشد.

	1	2	3	4	5	6	7	8	9	10
Key	33	172	9	55	489	958	39	603	802	971
Link										

ب) وضعیت لیست پس خاتمه اولین تکرار حلقه Main Loop چیست؟

	1	2	3	4	5	6	7	8	9	10
Key										
Link										

ج) به جای توضیح // insert printing code کدی بنویسید که در انتهای تابع کلیدهای لیست را به صورت مرتب شده صعودی چاپ کند.