



مبانی برنامه‌سازی (C) ۴۰۱۵۳
Fundamental of Programming (C)

دانشگاه صنعتی شریف
دانشکده مهندسی کامپیوتر

تاریخ آزمون: سه‌شنبه ۲۳ دی ۱۳۹۳

وقت: ۱۸۰ دقیقه

آزمون پایان ترم

به نکات زیر توجه داشته باشید.

نکته ۱: می‌توانید به ۱۱ سوال تستی جواب دهید و در صورت درست بودن ۱۰ سوال، نمره کامل سوالات تستی را دریافت خواهید کرد.

نکته ۲: سوالات تستی نمره منفی ندارد.

نکته ۳: جواب سوالات تستی را به ترتیب با ذکر شماره سوال و گزینه درست در پاسخ‌نامه بنویسید.

نکته ۳: کدهای خود را تمیز و خوانا نوشته و کامنت‌گذاری نمائید.

نکته ۴: سوال ۴ (سوال تشریحی آخر) مخصوص دانشجویان مهندسی کامپیوتر و برق بوده و پاسخ دادن به این سوال برای این دانشجویان الزامی می‌باشد. دانشجویان رشته‌های دیگر در صورت پاسخ به این سوال، نمره اضافی دریافت نخواهند کرد.

سوالات تستی (۵۰ نمره)

۱. خروجی برنامه زیر کدام گزینه است؟

```
int main()
{
    char *s= "hello";
    char *p = s;
    printf("%c\t%c", *(p + 3), s[1]);
    return 0;
}
```

- الف. h e
- ب. h l
- ج. h o
- د. l e

۲. خروجی برنامه زیر چیست؟

```
int main()
{
    int a[3] = {1, 2, 3};
    int *p = a;
    printf("%p\t%p", p, a);
    return 0;
}
```

الف. دو آدرس یکسان چاپ می‌شود.
 ب. دو آدرس متفاوت چاپ می‌شود.
 ج. خطای کامپایلر دارد.
 د. هیچکدام

۳. خروجی برنامه زیر کدام گزینه خواهد بود؟

```
void foo(int p[4])
{
    int i = 10;
    p = &i;
    printf("%d ", p[0]);
}

int main()
{
    int ary[4] = {1, 2, 3, 4};
    foo(ary);
    printf("%d ", ary[0]);
    return 0;
}
```

الف. 10 10
 ب. 10 1
 ج. 10 4
 د. Compile time error

۴. آرایه زیر داده شده است. خروجی برنامه به ازاء `printf("%c", books[2][5])` چه خواهد بود؟

```
char books[][40]={
    "C How to Program",
    "Fundamental of Physics ",
    "Calculus I",
    " The C Programming Language"
};
```

الف. u
 ب. a
 ج. l
 د. m

۵. خروجی کد زیر کدام گزینه است؟

```
int main()
{
    int a[] = {1, 2, 3, 4, 5};
    int b[4][4] = {{1, 2, 3, 4}, {5, 6, 7, 8},
                  {9, 10, 11, 12}, {13, 14, 15, 16}};
    printf("%d%d%d%d", *(a+3), (*(b+3)+2), a[2], b[2][1]);
    return 0;
}
```

الف. 415415
ب. 415310
ج. 310310
د. 310415

۶. خروجی برنامه زیر کدام گزینه خواهد بود؟

```
int a = 3;
int main()
{
    for(int i = 0; i < 2; i++){
        int a = 6;
        ::a++;
        printf("%d ", a);
        if(i == 1){
            int a = 5;
            ::a++;
            printf("%d ", a); } }
    printf("%d ", a);
    return 0;
}
```

الف. 7 8 6 3
ب. 7 7 5 3
ج. 6 6 5 3
د. 6 6 5 6

۷. خروجی برنامه زیر کدام گزینه خواهد بود؟

```
int main()
{
    char arr[]="ABCDEFGHijkl";
    int *fptr;
    fptr=(int *)arr;
    fptr++;
    printf("%s", fptr);
    return 0;
}
```

الف. ABCDEFGHIJKL
ب. CDEFGHIJKL
ج. EFGHIJKL
د. IJKL

۸. با فرض اینکه arr از محل 1000 حافظه شروع شود، خروجی برنامه زیر کدام گزینه خواهد بود؟

```
int main()
{
    int arr[2][2][2] = {1, 2, 3, 4, 5, 6, 7, 8};
    int *p, *q;
    p = &arr[1][0][1];
    q = arr[0][1];
    printf("%d %d\n", *p, *q);
    return 0;
}
```

الف. 6 3
ب. 8 3
ج. 6 1008
د. 8 1008

<pre>int g(int a, int b) { int x=12; return x*b; } int x=5; int main() { int i=20, j=10, p=0; p=g(i, j); printf("%d", p+x); return 0; }</pre>	<p>۹. خروجی برنامه زیر کدام گزینه خواهد بود؟</p> <p>الف. 132</p> <p>ب. 125</p> <p>ج. 55</p> <p>د. 0</p>
---	---

۱۰. با فرض ۳۲ بیتی بودن کامپایلر، خروجی برنامه زیر کدام گزینه خواهد بود؟

<pre>int main() { printf("%d %d\n", sizeof(NULL), sizeof("")); return 0; }</pre>	<p>الف. 1 1</p> <p>ب. 1 4</p> <p>ج. 4 1</p> <p>د. 4 4</p>
--	---

<pre>int main() { int i1; double d1; union { int i2; double d2; }; i1 = 5; i2 = 10; d1 = 5.5; d2 = 10.5; cout<<d1<<endl; cout<<d2<<endl; cout<<i1<<endl; cout<<i2<<endl; return 0; }</pre>	<p>۱۱. کدامیک از گزینه‌های زیر اشتباه است؟</p> <p>الف. مقدار 5.5 برای d1 چاپ می‌شود.</p> <p>ب. مقدار 10.5 برای d2 چاپ می‌شود.</p> <p>ج. مقدار 5 برای i1 چاپ می‌شود.</p> <p>د. مقدار 10 برای i2 چاپ می‌شود.</p>
--	--

سوالات تشریحی (۵۰ نمره)

سوال ۱. (۱۰ نمره)

می‌خواهیم با استفاده از n مکعب یک هرم بسازیم (n یک عدد صحیح مثبت است). ساخت هرم به این ترتیب است که در طبقه اول آن ۱ مکعب، در طبقه دوم آن ۲ مکعب، در طبقه سوم آن ۳ مکعب و به همین ترتیب در طبقه i مکعب وجود دارد. برنامه‌ای بنویسید که با دریافت عدد n حداکثر ارتفاع هرم را محاسبه کند.

نمونه ورودی	نمونه خروجی
۱	۱
۵	۳
۱۰	۴
۱۰۰	۱۴

سوال ۲. (۱۰ نمره)

برنامه‌ای بنویسید که عدد N را از کاربر دریافت کند و به تعداد N اسامی و تاریخ تولد N نفر را از کاربر دریافت کند و در آرایه‌ای از `Struct` ذخیره کند. سپس تاریخ امروز را دریافت کرده و اسامی کسانی را که تاریخ تولدشان امروز یا به فاصله ۱۰ روز کمتر از امروز رخ می‌دهد (۱۰ روز مانده به تاریخ تولد) را در خروجی چاپ کند. (راهنمایی: می‌توانید یک تابع بنویسید که ماه و روز را به معادل تعداد روزها تبدیل کند. مثال ششم اردیبهشت ۳۷ روز از اول سال فاصله دارد.)

سوال ۳. (۱۰ نمره)

تابع اکرم (Ackermann) با تعریف زیر را در نظر بگیرید.

$$A(m, n) = \begin{cases} n + 1 & \text{if } m = 0 \\ A(m - 1, 1) & \text{if } m > 0 \text{ and } n = 0 \\ A(m - 1, A(m, n - 1)) & \text{if } m > 0 \text{ and } n > 0 \end{cases}$$

تابعی بازگشتی بنویسید که با دریافت دو عدد m و n مقدار $A(m, n)$ را محاسبه کند. فرض کنید که اعداد ورودی از نوع صحیح بوده و خروجی نیز می‌تواند در `long int` ذخیره شود (فرض می‌کنیم ورودی‌ها اعداد کوچک می‌باشند).

سوال ۴. (۲۰ نمره)

تعریف زیر را برای خانواده در نظر بگیرید:
خانواده مجموعه ای از اشخاص هست شامل:

- ۱ مادر
- ۱ پدر
- ۱ یا چند فرزند (اعم از دختر و پسر)
- ۰ یا ۱ یا ۲ مادر بزرگ
- ۰ یا ۱ یا ۲ پدر بزرگ

و حداکثر اعضای یک خانواده ۱۰ نفر می باشد.

می خواهیم اعضای یک خانواده را با استفاده از آرایه ای از اشخاص مدل کنیم.
برای ذخیره سازی اطلاعات یک شخص از struct person استفاده می شود.

اعضای struct person شامل موارد زیر می باشد:

- نام کوچک
- جنسیت (لازم است جنسیت را با استفاده از enum با نام sex و با اعضای FEMALE و MALE بیان کنید)
- مادر (در صورت وجود مادر در خانواده برای آن شخص)
- پدر (در صورت وجود پدر در خانواده برای آن شخص)
- تعداد اعضای خانواده

الف) struct person را تعریف کنید.

```
struct person{  
...  
};
```

ب) تابعی بنویسد که با دریافت اطلاعات یک خانواده به عنوان پارامتر ورودی نام مادر بزرگ ها را در صورت وجود، در خروجی استاندارد چاپ نماید.

```
void printGrandMothers(struct person family[]);
```

ج) تابعی بنویسید که با دریافت اطلاعات یک خانواده و اندیس شخص مورد نظر به عنوان پارامتر ورودی نام برادرهای آن شخص را در صورت وجود، در خروجی استاندارد چاپ نماید.

```
void printBrothers(struct person family[], int personIndex);
```

د) تابعی بنویسید که با دریافت اطلاعات یک خانواده و نام کوچک و جنسیت به عنوان پارامتر ورودی، در صورت وجود ظرفیت (تعداد اعضای خانواده کمتر از 10 باشد) یک فرزند را به اعضای خانواده مقاردهی (اضافه) کند. توجه داشته باشید که در این جا اشخاص (اعضای خانواده) همواره از اندیس 0 تا اندیس (1 - تعداد اعضای خانواده) مقاردهی می شوند.

```
void addChild(struct person family[], char* name, enum sex s);
```

سوال 5. مخصوص دانشجویان مهندسی کامپیوتر و برق. (۲۰ نمره)

مرتب‌سازی سریع (Quick sort)

فرض کنید آرایه ای از اعداد صحیح نامرتب به صورت $A[n]$ داشته باشیم. در واقع $A[0], A[1], \dots, A[n-1]$ به دنباله‌ای از اعداد صحیح اشاره می‌کنند که مرتب نمی‌باشند. می‌خواهیم الگوریتمی برای مرتب‌سازی این آرایه بنویسیم. در این الگوریتم، ابتدا عنصری از آرایه به صورت تصادفی به عنوان نقطه اتکاء (pivot) انتخاب می‌شود و سپس بر اساس نقطه اتکاء آرایه به دو دسته‌ی کوچکتر یا مساوی با عنصر نقطه اتکاء و بزرگتر از عنصر نقطه اتکاء افراز (پارتیشن) می‌شود. به عبارت دیگر اگر عنصر $A[j]$ به عنوان عنصر نقطه اتکاء انتخاب شود ($0 \leq j \leq n-1$) و جایگاه واقعی این عنصر در آرایه مرتب شده خانه‌ی شماره p باشد (وقتی چند عنصر تکراری مانند عنصر نقطه اتکا داریم اندیس p برابر است با بزرگترین اندیسی که در آرایه‌ی مرتب شده عنصری با مقدار نقطه اتکا را شامل می‌شود). عمل افراز، آرایه را به دو قسمت، $A[0], \dots, A[p-1]$ که از عنصر نقطه اتکاء کوچکتر یا مساوی‌اند و $A[p+1], \dots, A[n-1]$ که از عنصر نقطه اتکاء بزرگترند تقسیم می‌کند. بعد از عمل افراز، دو دسته ایجاد شده به صورت بازگشتی مرتب می‌شوند که با قرار دادن نقطه اتکاء در بین آنها به آرایه مرتب‌شده می‌رسیم. به این الگوریتم مرتب‌سازی سریع (Quick Sort) می‌گویند که می‌توان آن را به صورت مراحل زیر نوشت.

- اگر تعداد عناصر آرایه بیشتر از یک است کارهای زیر را انجام بده:
- عنصری از آرایه را به صورت تصادفی (یا با هر مکانیزم دیگر) به عنوان عنصر نقطه اتکاء انتخاب کن.
- عناصر آرایه را بر اساس عنصر نقطه اتکاء به دو دسته‌ی کوچکتر یا مساوی عنصر نقطه اتکاء و بزرگتر از عنصر نقطه اتکاء تقسیم کن (توجه کنید که این مرحله، باعث می‌شود عنصر نقطه اتکا در مکان صحیح خود در آرایه مرتب شده قرار گیرد).
- به صورت بازگشتی دو دسته‌ی به دست آمده از افراز را مرتب کن.

تابعی بنویسید که الگوریتم مرتب‌سازی سریع را پیاده‌سازی نماید. فرض کنید تابعی با prototype زیر داریم که آرایه‌ی نامرتب A ، اندیس ابتدا (lo) و اندیس انتهای (hi) ورودی‌های آن هستند. شما کد مربوط به مرتب‌سازی را بنویسید.

```
void quickSort(int A[], int lo, int hi)
{
/* Your code */
}
```

در الگوریتم مرتب سازی سریع با دو مسئله روبه رو هستیم.

(۱) نحوه‌ی انتخاب Pivot که توسط تابع selectPivot انجام می‌شود. برای پیاده سازی تابع selectPivot کافی است یک عدد تصادفی بین lo و hi انتخاب کنید. Prototype آن به صورت زیر است.

```
int selectPivot(int lo , int hi);
```

(۲) نحوه افزایش کردن آرایه: یکی متداول‌ترین روش‌های پیاده‌سازی افزایش درجا است. در این روش از دو اشاره‌گر استفاده می‌شود. یکی از ابتدای آرایه و دیگری از انتهای آن شروع به حرکت می‌کنند. اشاره‌گر ابتدای آرایه از ابتدای آرایه شروع کرده و هر عنصر آرایه را با عنصر نقطه اتکاء مقایسه می‌کند. در صورت کوچکتر یا مساوی بودن عنصر با عنصر نقطه اتکاء به عنصر بعدی می‌رود در غیر این صورت حرکت نمی‌کند و به آن عنصر اشاره می‌کند. اشاره‌گر انتهای آرایه از انتها شروع کرده و عنصر آرایه را با عنصر نقطه اتکاء مقایسه می‌کند در صورت بزرگتر بودن عنصر به عنصر قبلی رفته در غیر این صورت حرکت نمی‌کند و به آن عنصر اشاره می‌کند. در حلقه‌ی افزایش وقتی دو اشاره‌گر ثابت شدند عناصری که به آن اشاره می‌کنند را با هم تعویض می‌کنیم و دوباره حرکت را ادامه می‌دهند. حلقه‌ی افزایش وقتی که دو اشاره‌گر به هم برسند به اتمام می‌رسد. توجه کنید که در پایان حلقه‌ی افزایش، مکانی که اشاره‌گر انتها به آن اشاره می‌کند مکان عنصر نقطه اتکاء در آرایه‌ی مرتب شده است. پس باید عنصر نقطه اتکاء را به آن جا انتقال دهید. توجه داشته باشید از آنجا که ممکن است مکان عنصر نقطه اتکاء عوض شود در ابتدای تابع افزایش آن را با عنصر اول آرایه در حال افزایش شدن جابه جا کنید تا بتوانید در انتها راحت‌تر به جای مناسب آن انتقال دهید.

موفق باشید.