

2024年程序设计与思维实践期末考试代码能力削弱考试（回忆版）

严正声明：本回忆版试卷所有内容均非回忆者本义，部分内容在不影响考察内容的情况下进行适当修正，请各位同学千万千万不要像这样写代码

本卷满分 100 分，考试时间 120 分钟。每题具体分配忘了

整理：[24学术讨论群](#)（[syhanjin](#) 老汉 离谱 潜伏 混子 浮萍 東牆 天赐 [卡基米](#) 黄鹂 Yasumi Speculator Schwarz Fun10165 Jaaack）以及 [24小登对个答案](#) 的群友们

感谢全体为回忆试卷做出贡献的群友们！

代码书写已按整理者习惯风格调整。

祝各位吃的开心！

一、单项选择题

1. `k` 为整形，`ptr1`，`ptr2` 为指向整形的指针。下列赋值错误的是：

- A. `k = *ptr1 + *ptr2;`
- B. `ptr1 = k;`
- C. `ptr1 = ptr2;`
- D. `k = *ptr1*(*ptr2);`

2.

```
struct cmplx {
    int x;
    int y;
} cnum[2] = { {1, 3}, {2, 6} };
```

`cnum[0].y*cnum[1].x` 的值为

- A. `0`
- B. `1`
- C. `3`
- D. `6`

3. 有下列程序，请问输出结果为多少：

```
#include <stdio.h>
int abc (int u, int v);
int main()
{
    int a = 24, b = 16, c;
    c = abc(a,b);
    printf("%d",c);
    return 0;
}
int abc (int u, int v)
{
    int w;
    while(v)
    {
```

```

        w = u % v;
        u = v;
        v = w;
    }
    return u;
}

```

- A. 1 B. 16 C. 8 D. 0

4. `struct Student` {
 `char name[100];`
 `// ...`
} `student[50];`

将数据通过 `fp` 写入文件的写法错误的是（假设文件已经正常打开）

```

/* A */fwrite(student, sizeof(struct Student), 50, fp);
/* B */fwrite(student, sizeof(student), 1, fp);
/* C */fwrite(student, 25 * sizeof(struct Student), 25, fp);
/* D */for(int i = 0; i < 50; i++) fwrite(student, sizeof(struct Student), 1, fp);

```

5. `A` 的 ASCII 码为 `65`，`char ch = 'A' + '6' - '3'` 则 `ch` 表示的字符是

- A. `D` B. `68` C. 不确定的值 D. `C`

6. 下面说法正确的是

- A. 在函数调用中，只能把实参的值传递给形参，不能把形参的值传递给实参
- B. 函数既可以嵌套定义又可以递归调用
- C. 所有函数必须有返回值
- D. 调用的函数必须包含在同一个源文件里面

7. 定义二维数组 `int a[m][n];`，以下表示 `a[i][j]` 的方式正确的是

- A. `*(a+i*n+j)`
- B. `*(a+j*n+i)`
- C. `*(*(a+i)+j)`
- D. `*(a+i+j)`

8. 有 `int a, b, k, c1, c2`，下列 `switch` 语句使用正确的是

```

/* A */
switch (a + b)
{
    case 1: k+=x,x=y; break;
    case 0: a=1,b=2; break;
    default: :c1=c2,x=y; break;
}
/* B */
switch a
{
    case c1: a=c1,b=c2; break;
    case c2: k+=x*y; break;
}

```

```

/* C */
switch (a*k)
{
    case 1.0f: a=b*7; break;
    case 2.0f: c1+=c2; break;
}
/* D */
switch (x*2+y*11)
{
    case 3: case 4: a*=7; x++; break;
    case 10: case 11:y=x+a; b*=2; break;
    default: ...
}

```

9. 下列进制转换错误的是

- A. $(1101)_{-2} = (13)_{+10}$ B. $(1101)_{-3} = (37)_{+10}$ C. $(1101.1)_{-2} = (13.5)_{+10}$ D. $(1101.1)_{-3} = (13.2)_{+10}$

10. `int *f();` 下列对 `f` 解释正确的是:

- A. `f` 是元素为指针的数组
 B. `f` 是指向数组的指针
 C. `f` 指向一个函数
 D. `f` 是一个返回值为整型指针的函数

二、填空题

1. 设计程序证明 n^3 可以拆分成连续奇数的和

注: 本题本身具有的问题 (如 `n=1` 时无法计算) 未修复

```

#include <stdio.h>
int main()
{
    long int n, i, j, sum, k;
    scanf("%ld", &n);
    k = n * n * n;
    for (i = 1; i < k / 2; i += 2)
    {
        for (j = i, sum = 0; ___(1)___; j += 2)
            sum += j;
        if ( ___(2)___ )
            printf("%ld*%ld*%ld=%ld=from %ld to %ld.\n", n, n, n, sum, i, ___(3)___ );
    }
    return 0;
}

```

2. 用链表实现反向输出字符串

注: 本题目 `malloc` 没有 `free` 和 `#include<stdlib.h>`、未检查内存是否分配成功等问题已修复

```

#include <stdio.h>

```

```

#include <stdlib.h>
struct node {
    char data;
    struct node* next;
} head;
int main()
{
    int ch;
    struct node* p;
    head = NULL;
    while ((ch=getchar()) != '\n')
    {
        p = ____(1)___ ;
        if (p == NULL) {...} // 略去
        p -> next = head;
        ____(2)___ ;
    }
    ____(3)___ ;
    struct node* temp;
    while (p != NULL)
    {
        printf("%c", p->data);
        temp = p;
        p = p->next;
        free(temp);
    }
    return 0;
}

```

3. `a[3][3]` 里面找最大值和对应的行指标（主函数已略去）

```

#define N 3
#define M 3
int f(int a[N][M], int *n)
{
    int i, j, row = 1, column = 1;
    for (i = 0; i < N; i++)
        for (j = 0; j < M; j++)
            if (a[i][j] > a[row][column]) row = i, column = j;
    *n = ____(1)___ ;
    return ____(2)___ ;
}

```

三、简答题

1. 什么是递归函数？递归函数的两个基本要素是什么？简述递归函数的优点和缺点
2. 请从调用范围、存储位置、生存时间分析全局变量、静态局部变量和动态变量的区别
3. (1). 请分析计算选择排序（选择最小值）和快速排序算法最好、最坏、平均三种情况的时间复杂度（待处理变量数为 `n`）
 - (2). 用数字 `12345` 写出两种排序方式的最坏时间排列

(3). 两种排序算法是否稳定? 若不稳定, 举出反例 (可以用 `1 2 2'` 来说明)

4. (1). 简述单向链表和数组的优缺点

(2). 对于单向链表, 双向链表和循环链表, 只给出当前位置指针 `p` 能否找到它的上一级位置, 并计算时间复杂度

(3).

注: 本题配图略

```
struct node {
    int data;
    struct node *next;
};
struct link {
    struct node *head;
};
```

设计创建循环链表的构造函数 (包含一个数据无效的 `head` 节点)

```
// 创建成功返回 0, 创建失败返回 -1
int construct(struct link *lnk);
```

四、程序设计

1. 给出一段字符 `T[]` (如 `"ABBA"`), 判断这段字符能否完全相消, 如果能则打印 `"You win!\n"`, 不能则打印 `"You Lose!\n"`

注: 本题有一大堆情境题干, 忘了。完全相消 类似于消消乐, 相邻的相同字符相互消去

题干中已给出栈的结构体定义和四个栈相关函数:

```
void init(stack *s); // 初始化栈顶指针
int empty(stack *s); // 判断栈是否空, 空返回 1, 非空返回 0
int push(stack *s, char e); // 将一个元素入栈, 成功返回 1, 失败返回 0
int pop(stack *s, char *e); // 弹出栈顶元素, 并使 e 指向被弹出的元素, 成功返回 1, 失败返回 0
```

可以直接调用这四个函数

设计函数 `void judge(char T[], int n);` 实现

2. 用递归计算 $f(x, n) = x - x^2 + x^3 + \dots + (-1)^{n-1}x^n$

函数原型为 `double fn(double x, int n);`

(1) 写出 $f(x, n)$ 的递归公式

(2) 设计函数实现 `fn` 的功能 (递归起始为 `n=1`)

(3) 写出一个包括头文件, `main` 函数, 调用函数的完整程序