

上海市高等教育自学考试
计算机信息管理专业（独立本科段）
《软件开发工具》上机考核大纲

一、考核目标

熟悉 Eclipse cpp + MinGW 的安装、配置；能使用 eclipse 编程环境，完成一个 C 项目的创建、编辑、调试到运行的生命周期过程。

二、考核要求

实现 Eclipse cpp + MinGW 的安装、配置，C 项目运行的输出与试题的要求基本一致。

三、软件环境

操作系统：Windows XP。

四、考核方式

闭卷考试，用时一个半小时。每个考生按试题的要求安装、配置开发环境 eclipse，完善一个 C 程序。

五、考核范围

- Eclipse Galileo + MinGW 5.1.6 集成开发环境的安装、配置、使用：新建工程、编辑 C 程序、编译文件与项目、调试程序、运行程序。
- 数组、线性表、栈、队列和串等数据结构上的基本算法，数组、线性表、栈、队列和串等的简单应用，选择排序、冒泡排序、插入排序、基排序、快速排序及其应用；顺序查找、二分法查找及其应用。
- 程序输出到文件，从文件读入并处理。

六、结果提交要求

考生在运行 eclipse 时，设置工作空间为 C:\temp\workspace，在 C:\temp\workspace 下创建工程（目录），考试时，将源程序文件，以及系统产生的工作文件都保存在这个工程目录之下。考试结束时，将工程目录下的源程序文件和输出文件提交到服务器。

要求源程序文件按以下格式命名：**试题代码考号.c**，输出文件名为 **output.txt**。

其中考号是考生参加本课程上机考试的准考证号，试题代码是上机考试时使用的试题代码。例如，某考生的考号为 2012001，试卷代码为 B，则源程序文件名为：B2012001.c。

七、试题形式

给定系统软件 eclipse-cpp-galileo-SR2-win32 和 MinGW516。

给定试题要求以及实现要求的源程序的部分源代码。

八、例题

给定一个程序和一个函数，程序的功能是按要求自动生成数列并排序、函数的功能是在数列中查找 3 的倍数。

要求：合成一个程序，其功能是：

- 显示"请选择元素个数（大于 0 且小于 1001），以及数的范围："

- 自动生成数列。
- 显示"请选择功能，f 表示查询、s 表示排序"。
- 按要求运行。
- 程序的输出也写入文件 output.txt

排序程序源代码：

```
#include <stdio.h> /* 输入输出库函数的头文件 */
#include <time.h> /* 时间库函数的头文件 */
#include <stdlib.h> /* 数学或内存分配库函数的头文件 */
void initArray(int a[], int n, int r){
    int k;
    long now;
    srand(time(&now)); /* 用时间初始化随机数发生函数的初态，使初态总不相同 */
    for(k = 0; k < n; k++) /* 产生n个0..r-1以内的随机数 */
        a[k] = rand() % r; /* 调用随机函数 */
}

void writeArray(int a[], int n, int c){
    int k;
    for(k = 0; k < n; k++){
        if(k % c == 0)
            printf("\n\t\t");
        printf("%6d", a[k]);
    }
    printf("\n");
    fflush(stdout);
}

void Sort(int * e, int n){
    int i, j, temp;
    for(i = 1; i < n; i++) {
        temp = e[i];
        for(j=i-1; j>=0&&e[j]>temp; j--)
            e[j+1] = e[j];
        e[j+1] = temp;
        printf("i = %d\t\t", i);
        writeArray(e, n, 10);
        getchar();
    }
}

int * r;
int main(void){
    int n, range;
    while (1){
        printf("请选择元素个数（大于0且小于1001），以及数的范围:\t");
        fflush(stdout);
        scanf("%d%d", &n, &range);
        if (n>0 && n<1001)
            break;
    }
    fflush(stdin);
    r = (int *)malloc(n*sizeof(int));
```

```
    initArray(r, n, range);
    printf("原序列: \t\t");
    writeArray(r, n, 10);
    Sort(r, n);
    printf("排序后的序列: \t");
    writeArray(r, n, 10);
    getchar();
    return 0;
}
```

查询 3 的倍数函数源代码：

```
FILE *fp;
void fine(int * e, int n){
    int i, j=0;
    for (i = 1; i < n; i++) {
        if (!(e[i]%3)){
            if (j % 10){
                printf("\n\t\t");
                fprintf(fp, "\n\t\t");
            }
            printf("%6d", e[i]);
            fprintf(fp, "%6d", e[i]);
            j++;
        }
    }
    printf("\n\t\t共有 %d 个 3 的倍数", j);
    fprintf(fp, "\n\t\t共有 %d 个 3 的倍数", j);
}

int * r;
int main(void){
    int n, range;
    while (1){
        printf("请选择元素个数（大于0且小于1001），以及数的范围:\t");
        fflush(stdout);
        scanf("%d%d", &n, &range);
        if (n>0 && n<1001)
            break;
    }
    fflush(stdin);
    fp = fopen("output.txt", "w+"); /* 写方式打开文件 */
    r = (int *)malloc(n*sizeof(int));
    initArray(r, n, range);
    printf("原序列: \t");
    fprintf(fp, "原序列: \t");
    writeArray(r, n, 10);
    printf("其中3的倍数: \t");
    fprintf(fp, "其中3的倍数: \t");
    fine(r, n);
    fflush(stdout);
    return 0;
}
```

参考答案：

```
#include <stdio.h> /* 输入输出库函数的头文件 */
#include <time.h> /* 时间库函数的头文件 */
#include <stdlib.h> /* 数学或内存分配库函数的头文件 */
FILE *fp;
void print(char *s){
    printf(s);
    fprintf(fp, s);
}

void initArray(int a[], int n, int r){
    int k;
    long now;
    srand(time(&now)); /* 用时间初始化随机数发生函数的初态，使初态总不相同 */
    for(k = 0; k < n; k++) /* 产生n个0..r-1以内的随机数 */
        a[k] = rand() % r; /* 调用随机函数 */
}

void writeArray(int a[], int n, int c){
    int k;
    for(k = 0; k < n; k++){
        if(k % c == 0)
            print("\n\t\t");
        printf("%6d", a[k]);
        fprintf(fp, "%6d", a[k]);
    }
    print("\n");
    fflush(stdout);
}

void Sort(int * e, int n){
    int i, j, temp;
    for(i = 1; i < n; i++) {
        temp = e[i];
        for(j=i-1; j>=0&&e[j]>temp; j--)
            e[j+1] = e[j];
        e[j+1] = temp;
        printf("i = %d\t\t", i);
        fprintf(fp, "i = %d\t\t", i);
        writeArray(e, n, 10);
        getchar();
    }
}

void fine(int * e, int n){
    int i, j=0;
    for (i = 1; i < n; i++) {
        if (!(e[i]%3)){
            if (j % 10)
                print("\n\t\t");
        }
    }
    printf("\n\t\t共有 %d 个 3 的倍数", j);
    fprintf(fp, "\n\t\t共有 %d 个 3 的倍数", j);
}

int * r;
int main(void){
    int n, range;
    char ch;
    while (1){
        printf("请选择元素个数（大于0且小于1001），以及数的范围:\t");
        fflush(stdout);
        scanf("%d%d", &n, &range);
        if (n>0 && n<1001)
            break;
    }
    fflush(stdin);
    fp = fopen("output.txt", "w+"); /* 写方式打开文件 */
    r = (int *)malloc(n*sizeof(int));
    initArray(r, n, range);
    while (1){
        printf("请选择\n\t\t排序s\t\t查询f \t");
        fflush(stdout);
        scanf("%c", &ch);
        if (ch=='f' || ch=='s')
            break;
    }
    fflush(stdin);
    print("原序列: \n\t");
    writeArray(r, n, 10);
    if (ch=='s'){
        Sort(r, n);
        print("排序后的序列: \t");
        writeArray(r, n, 10);
    }else{
        printf("其中3的倍数: \t");
        fprintf(fp, "其中3的倍数: \t");
        fine(r, n);
        fflush(stdout);
    }
    return 0;
}
```