

1. 编写程序打印99乘法口诀表。

2. 请编写程序将存放10个整数的数组颠倒顺序。<br> 12 3 94 7 21 5 0 18 33 73 变成<br> 73 33 18 0 5 21 7 94 3 12

3. 以结构体的方式来定义几何中的点、圆形和正方形，要求：<br> a. 编写函数 `isInCircular(struct circular c, struct point p)`，判断点p是否在圆形c内部；<br> b. 编写函数 `isInRectangle(struct rectangle c, struct point p)`，判断点p是否在矩形c内部；<br>

1. 编写程序打印99乘法口诀表。

```
void main( int argc, char *argv[] ) {
    static const char *optString = ":l:yt::";
    int opt;
    opt = getopt( argc, argv, optString );
    while ( opt != -1 ) {
        switch ( opt ) {
            case 'l':
                printf("input opt is:%c\n", opt);
                printf("the parms is:%s\n", optarg) ;
                break;
            case 'y':
                printf("input opt is:%c\n", opt);
                break;
        }
        opt = getopt( argc, argv, optString );
    }
}
```

2. 请编写程序将存放10个整数的数组颠倒顺序。<br> 12 3 94 7 21 5 0 18 33 73 变成<br> 73 33 18 0 5 21 7 94 3 12

```
docker build -t mynodeapp .
docker build -t mynodeapp .
docker build -t mynodeapp .
docker build -t mynodeapp .
docker build -t mynodeapp .
```

3. 以结构体的方式来定义几何中的点、圆形和正方形，要求：<br> a. 编写函数 `isInCircular(struct circular c, struct point p)`，判断点p是否在圆形c内部；<br> b. 编写函数 `isInRectangle(struct rectangle c, struct point p)`，判断点p是否在矩形c内部；<br>

FROM hub.c.163.com/nc2/nodejs:0.12.2

```
# Create app directory
RUN mkdir -p /home/Service
WORKDIR /home/Service
```

```
# Bundle app source
COPY . /home/Service
RUN npm install

EXPOSE 8888
CMD [ "npm", "start" ]
awfaffa
awfafaf
```

考生学号：2016210405062

试卷成绩：58

---

1. 编写程序打印99乘法口诀表。

```
#include <stdio.h>
int main( int argc, char *argv[] )
{
    if ( argc == 2 ) {
        printf("The argument supplied is %s\n", argv[1]);
    } else if ( argc > 2 ) {
        printf("Too many arguments supplied.\n");
    } else {
        printf("One argument expected.\n");
    }

    printf("%d",i);
}
```

2. 请编写程序将存放10个整数的数组颠倒顺序。<br>12 3 94 7 21 5 0 18 33 73 变成<br>73 33 18 0 5 21 7 94 3 12

```
$ ./main -l aaa
input opt is:l
the parms is:aaa

$ ./main -laaa
input opt is:l
the parms is:aaa

$ ./main -l 4343
input opt is:l
the parms is:4343

$ ./main -y
input opt is:y

$ ./main -y 444
input opt is:y

$ ./main -t aaa
input opt is:t
the parms is:(null)

$ ./main -twaaa
input opt is:t
```

```

the parms is:waaa

$ ./main -m
invaild opt!

$ ./main -l
Lack of option parameters!

$ ./main -t
input opt is:t
the parms is:(null)

$ ./main -yl 34 -t liyang
input opt is:y
input opt is:l
the parms is:34
input opt is:t
the parms is:(null)

$ ./main -yl 34 -tliyang
input opt is:y
input opt is:l
the parms is:34
input opt is:t
the parms is:liyang

```

3. 以结构体的方式来定义几何中的点、圆形和正方形，要求：<br> a. 编写函数 **isInCircular(struct circular c, struct point p)**，判断点p是否在圆形c内部；<br> b. 编写函数 **isInRectangle(struct rectangle c, struct point p)**，判断点p是否在矩形c内部；<br>

```

#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>

void main( int argc, char *argv[] ) {
    static const char *optString = ":l:yt::";
    int opt;
    opt = getopt( argc, argv, optString );
    while ( opt != -1 ) {
        switch ( opt ) {
            case 'l':
                printf("input opt is:%c\n", opt);
                printf("the parms is:%s\n", optarg) ;
                break;
            case 'y':
                printf("input opt is:%c\n", opt);
                break;
            case 't':
                printf("input opt is:%c\n", opt);

```

```

        printf("the parms is:%s\n", optarg) ;
        break;
    case '?':
        printf("invaild opt!\n");
        break;
    case ':':
        printf("Lack of option parameters!\n");
        break;
    }
    opt = getopt( argc, argv, optString );
}
}

```

考生学号：2016210405065

试卷成绩：

1. 编写程序打印99乘法口诀表。
2. 请编写程序将存放10个整数的数组颠倒顺序。<br> 12 3 94 7 21 5 0 18 33 73 变成<br> 73 33 18 0 5 21 7 94 3 12
3. 以结构体的方式来定义几何中的点、圆形和正方形，要求：<br> a. 编写函数 `isInCircular(struct circular c, struct point p)`，判断点p是否在圆形c内部；<br> b. 编写函数 `isInRectangle(struct rectangle c, struct point p)`，判断点p是否在矩形c内部；<br>

考生学号：2016210405057

试卷成绩：0

1. 编写程序打印99乘法口诀表。
2. 请编写程序将存放10个整数的数组颠倒顺序。<br> 12 3 94 7 21 5 0 18 33 73 变成<br> 73 33 18 0 5 21 7 94 3 12
3. 以结构体的方式来定义几何中的点、圆形和正方形，要求：<br> a. 编写函数 `isInCircular(struct circular c, struct point p)`，判断点p是否在圆形c内部；<br> b. 编写函数 `isInRectangle(struct rectangle c, struct point p)`，判断点p是否在矩形c内部；<br>

考生学号：2016210405071

试卷成绩：0

1. 编写程序打印99乘法口诀表。
2. 请编写程序将存放10个整数的数组颠倒顺序。<br> 12 3 94 7 21 5 0 18 33 73 变成<br> 73 33 18 0 5 21 7 94 3 12
3. 以结构体的方式来定义几何中的点、圆形和正方形，要求：<br> a. 编写函数 `isInCircular(struct circular c, struct point p)`，判断点p是否在圆形c内部；<br> b. 编写函数 `isInRectangle(struct rectangle c, struct point p)`，判断点p是否在矩形c内部；<br>

考生学号：2016210405051

试卷成绩：0

1. 编写程序打印99乘法口诀表。
2. 请编写程序将存放10个整数的数组颠倒顺序。<br> 12 3 94 7 21 5 0 18 33 73 变成<br> 73 33 18 0 5 21 7 94 3 12
3. 以结构体的方式来定义几何中的点、圆形和正方形，要求：<br> a. 编写函数 `isInCircular(struct circular c, struct point p)`，判断点p是否在圆形c内部；<br> b. 编写函数 `isInRectangle(struct rectangle c, struct point p)`，判断点p是否在矩形c内部；<br>

考生学号：2016210405079

试卷成绩：0

1. 编写程序打印99乘法口诀表。
2. 请编写程序将存放10个整数的数组颠倒顺序。<br> 12 3 94 7 21 5 0 18 33 73 变成<br> 73 33 18 0 5 21 7 94 3 12
3. 以结构体的方式来定义几何中的点、圆形和正方形，要求：<br> a. 编写函数 `isInCircular(struct circular c, struct point p)`，判断点p是否在圆形c内部；<br> b. 编写函数 `isInRectangle(struct rectangle c, struct point p)`，判断点p是否在矩形c内部；<br>

考生学号：2016210405075

试卷成绩：0

1. 编写程序打印99乘法口诀表。
2. 请编写程序将存放10个整数的数组颠倒顺序。<br> 12 3 94 7 21 5 0 18 33 73 变成<br> 73 33 18 0 5 21 7 94 3 12
3. 以结构体的方式来定义几何中的点、圆形和正方形，要求：<br> a. 编写函数 `isInCircular(struct circular c, struct point p)`，判断点p是否在圆形c内部；<br> b. 编写函数 `isInRectangle(struct rectangle c, struct point p)`，判断点p是否在矩形c内部；<br>

考生学号：2016210405076

试卷成绩：0

1. 编写程序打印99乘法口诀表。
2. 请编写程序将存放10个整数的数组颠倒顺序。<br> 12 3 94 7 21 5 0 18 33 73 变成<br> 73 33 18 0 5 21 7 94 3 12
3. 以结构体的方式来定义几何中的点、圆形和正方形，要求：<br> a. 编写函数 `isInCircular(struct circular c, struct point p)`，判断点p是否在圆形c内部；<br> b. 编写函数 `isInRectangle(struct rectangle c, struct point p)`，判断点p是否在矩形c内部；

<br>

考生学号： 2016210405070

试卷成绩：0

---

1. 编写程序打印99乘法口诀表。

2. 请编写程序将存放10个整数的数组颠倒顺序。<br> 12 3 94 7 21 5 0 18 33 73 变  
成<br> 73 33 18 0 5 21 7 94 3 12

3. 以结构体的方式来定义几何中的点、圆形和正方形，要求：<br> a. 编写函数  
isInCircular(struct circular c, struct point p)，判断点p是否在圆形c内部；<br> b. 编  
写函数isInRectangle(struct rectangle c, struct point p)，判断点p是否在矩形c内部；  
<br>