

## 第15回：小テスト（プログラミング試験） プログラミング及び実習 II

1

### ex14-1：自然数の和（再帰版）

```
#include <stdio.h>

int rsum(int n);

int main(void) {
    int n;
    do {
        scanf("%d", &n);
    } while (n < 1);

    printf("%d\n", rsum(n));
    return 0;
}

int rsum(int n) {
    if (n < 1)
        return 0;
    else if (n == 1)
        return 1;
    else
        return n + rsum(n-1);
}
```

3

## 第14回課題の解答例

2

### ex14-2：ユークリッドの互除法（再帰版）

```
#include <stdio.h>

int gcd(int, int);

int main() {
    int m, n, a;
    do {
        scanf("%d%d", &m, &n);
        a = gcd(m, n);
    } while (a < 0);

    printf("%d\n", a);
    return 0;
}

int gcd(int m, int n) {
    if (m<0 || n<0 || (m==0 && n==0)) return -1;
    if (m < n) {
        int tmp = m;
        m = n;
        n = tmp;
    }
    if (n == 0)
        return m;
    else
        return gcd(n, m%n);
}
```

4

## ex14-3：フィボナッチ数列（再帰版）

```
#include <stdio.h>

int fibonacci(int);

int main() {
    int n;
    scanf("%d", &n);
    printf("%d\n", fibonacci(n));
    return 0;
}

int fibonacci(int n) {
    if (n <= 0) return 0;
    else if (n == 1) return 0;
    else if (n == 2) return 1;
    else
        return fibonacci(n-1) + fibonacci(n-2);
}
```

5

## 第15回の内容

第15回 1/14 の小テスト（プログラミング試験）について

7

## ex14-4：フィボナッチ数列（メモ化再帰（動的計画法）版）

```
#include <stdio.h>
#define N 10000

long fibonacci2(int, long[]);

int main() {
    long fibo[N];
    for (int i=0; i<N; i++) fibo[i] = -1;

    int n;
    scanf("%d", &n);
    printf("%ld\n", fibonacci2(n, fibo));
    return 0;
}

long fibonacci2(int n, long fibo[]) {
    if (n <= 0) return 0;
    else if (n == 1) return 0;
    else if (n == 2) return 1;
    else if (fibo[n] == -1 || n >= N)
        fibo[n] = fibonacci2(n-1, fibo) + fibonacci2(n-2, fibo);
    return fibo[n];
}
```

6

## 小テスト（プログラミング試験）について

日時：1月14日（木）4-5講時（第15回）最大2時間

場所：オンライン（ネットワーク環境必須）

実施方法：

複数問題の選択制で、50点を超える点数は切り捨て（50点満点）

問題の提示と提出は Sharif-Judge（自動採点はされません）

指定方法（Awesome Screenshot）でのデスクトップ録画が必須

自分のソースファイル、ネットワーク検索、書籍資料の参照可

他者とのコミュニケーション（会話・チャット・SNS・ファイル送付など）不可

PCのデスクトップ画面・PCカメラ画像・音声録音は原則必須

8

## プログラミング試験に関するFAQ

選択問題はいくつ解答したらよいですか？

すべて答えても、一つも答えなくても構いません。各問の配点は Sharif-Judge 上で確認できます。提出されたものは全て採点され、その合計点が50点満点で成績に反映されます。

デスクトップ画面やカメラ画像などの録画に失敗した場合はどうなりますか？

受験したと認められません。受験途中で操作ミスなどにより録画を停止してしまった場合は、速やかに録画を再開し、録画リンクは試験時間中の全ての（複数の）URLリンクを提出すればOKです。特別な事情があると認められた場合は、考慮されることがあります。

自宅のネットワーク環境やPCが不安定で試験がうまく受けられるか心配です。

入構制限が無ければ、大学内の他の実習室などで受験することも可能です。自分のPCの動作に不安がある場合は、大学の実習室PCで動作する Ubuntu Linux を利用することができます。ただし、実習室PCでのデスクトップ録画のために、各自で事前の準備が必要です。また、コロナ感染状況によっては大学の実習室を使えなくなることがあります。学部のサポートルーム（1-443）などを利用して自宅PC環境の整備を進めてください。

試験中にTAからのサポートを受けることができますか？

PC操作や Sharif-Judge への提出を含め、試験中のTAサポートは受けられません。

## プログラミング試験に関するFAQ 2

カメラ動画や音声録音がどうしてもうまくいきません。

特別な事情があると認められる場合には、カメラ動画や音声録音が含まれない場合も受験として認められますので、事前に申し出て下さい。ただし、これらは不正行為が疑われたときの潔白性の証明となりますので、利用を強くお勧めします。

同一のソースファイル（コピーコード）であるという判定はどうやってなされますか？

ソースコードの記述の同一性をチェックした上で、デスクトップ録画でPC上でのプログラミングの過程を確認し、他人のコピーコードであるか、あるいは、自分で作成したものが偶然に他のコードと類似したものを判定します。

コピーコードが疑われた場合はどうなりますか？

上記でコピーコード認定されたものは不正行為として扱われます。また、疑わしいケースの場合は、後日、口頭試問を行い各問の可否を判定します。悪質なものは学則に則って処理します。

その他、試験に関する質問があれば「★質問チャンネル」へ