

第1回

第1回 統合開発環境の利用1 プロジェクトの管理, コードの記述 [↑]

本日の内容 [↑]

- はじめに
 - 就職とプログラミング
 - プログラミングが苦手
 - 2つの要因 (文法, 処理の流れ)
 - 1単語ずつ理解ようとしている? ぼーっと眺めていない?
 - コンパイルエラーのメッセージを読もうとしている?
- プログラミング言語I演習における『[読](#)』
- プログラミング言語I演習での評価
 - 結果より過程を重視
 - 初回コンパイル前のソースを記録
 - コンパイルエラーを記録
 - 積極性を重視
 - 相談、質問
 - 演習・課題へ独自の工夫・アイデアを追加
 - 教員, 技術職員, TAによるバックアップ
 - 「分かるまで教えます」
 - 「答えではなく, 考え方を教えます」
- 統合開発環境の使い方
 - [プロジェクトを作る](#)
 - [旧バージョンの場合](#)
 - コンパイルエラーを知る
- 演習
 - 演習1.1
 - 課題1.1
 - 課題1.2

[演習1.1] [↑]

コンパイルエラーの文章を読んでみよう.

手順 [↑]

1. 下記のソースを入力する. ひととおり入力を終わったら, **ビルドする前にソースをレポートに貼り付けておく.**
2. ビルドする. **ビルドエラーをレポートに貼り付けておく.**
3. エラーをひとつひとつ読み, 修正する.
4. エラーを全て取り除いた後, 実行して結果を確かめる.
5. エラーを取り除いたソースをレポートに貼り付ける.
6. 実行結果をレポートに貼り付ける.

ソース [↑]

```

#include <stdio.h>
#include <math.h>

int main(){

    int height;
    int lower = 150;
    int upper = 190;
    int step = 5;
    double weightStd;
    double weightIdeal;
    double weightModel;

    printf("身長 標準体重 美容体重 モデル体重\n\n");

    for (height = lower; height <= upper; height += step){
        weightStd = (height - 100) * 0.9;
        printf("%3dcm %2.2fkg ", height, weightStd);

        weightIdeal = pow(height / 100.0, 2.0) * 20;
        printf("%2.2fkg ", weightIdeal);

        weightModel = pow(height / 100.0, 2.0) * 19;
        printf("%2.2fkg\n", weightModel);
    }

    return 0;
}

```

[課題1.1] [±]

時刻ごとに異なるメッセージ（午前中なら「おはよう」、午後なら「こんにちは」など）を出力するプログラムを作成せよ。

手順 [±]

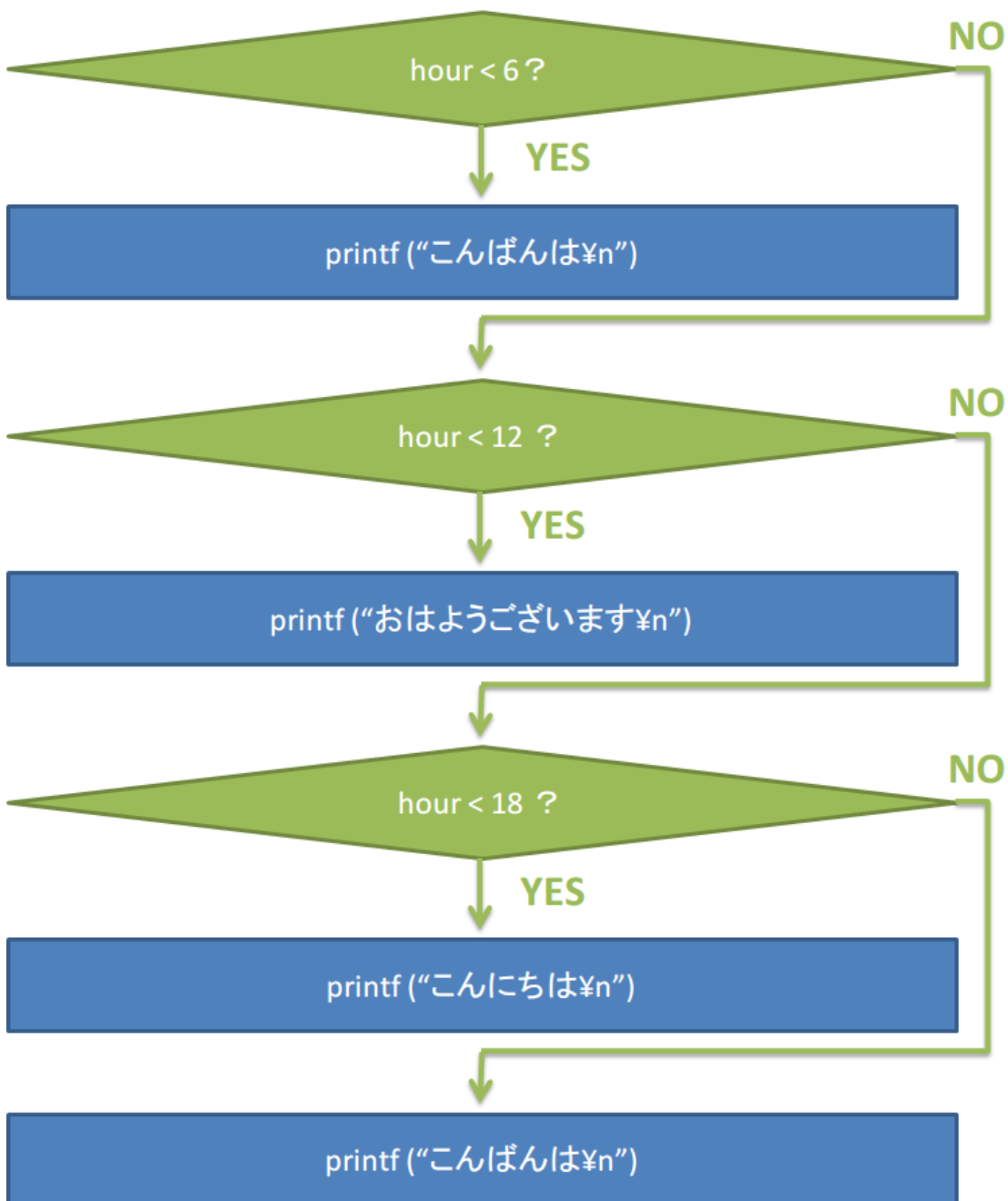
1. 下記のヒントをもとにプログラムを作成する。マスクしてある部分は自力で考える。
2. ひととおり入力を終わったら、**ビルドする前にソースをレポートに貼り付けておく**。
3. ビルドする。**ビルドエラーをレポートに貼り付けておく**。
4. エラーをひとつひとつ読み、修正する。
5. エラーを全て取り除いた後、実行して結果を確かめる。
6. 結果が正解と異なっていた場合は、ソースを見直す。
7. エラーを取り除いたソースをレポートに貼り付ける。
8. 実行結果をレポートに貼り付ける。
9. 友人と相談する場合も、まずは自力で考えること。誰と相談したか、また、相談した結果気がついた間違いをレポートに記述すること。

ヒント [±]

1. if文を用いること。

```
1 #define _CRT_SECURE_NO_WARNINGS
2
3 #include <stdio.h>
4 #include <time.h>
5
6 int main ( ) {
7
8     time_t timer = time (NULL); // time 関数を使って時刻を取得
9     struct tm *t; // tm構造体へのポインタ(時刻に関する要素を含む)
10    int hour; // 現在時刻の時
11    int min; // 現在時刻の分
12
13    // 現在時刻より時と分を取得
14    t = localtime(&timer);
15    hour = t->tm_hour;
16    min = t->tm_min;
17
18    printf("現在%d時%d分です\n", hour, min);
19
20
21
22
23
24
25
26
27 }
28 return 0;
29 }
```

メッセージ表示処理のフローチャート



[課題1.2] †

1から100までの数の和を求めるプログラムを作成せよ。

†

手順 †

1. 自分で考えながらソースを入力する。ひととおり入力を終わったら、**ビルドする前にソースをレポートに貼り付けておく**。
2. ビルドする。**ビルドエラーをレポートに貼り付けておく**。
3. エラーをひとつひとつ読み、修正する。
4. エラーを全て取り除いた後、実行して結果を確かめる。
5. 結果が正解と異なっていた場合は、ソースを見直す。
6. エラーを取り除いたソースをレポートに貼り付ける。
7. 実行結果をレポートに貼り付ける。
8. 友人と相談する場合も、まずは自力で考えること。誰と相談したか、また、相談した結果気がついた間違いをレポートに記述すること。

†

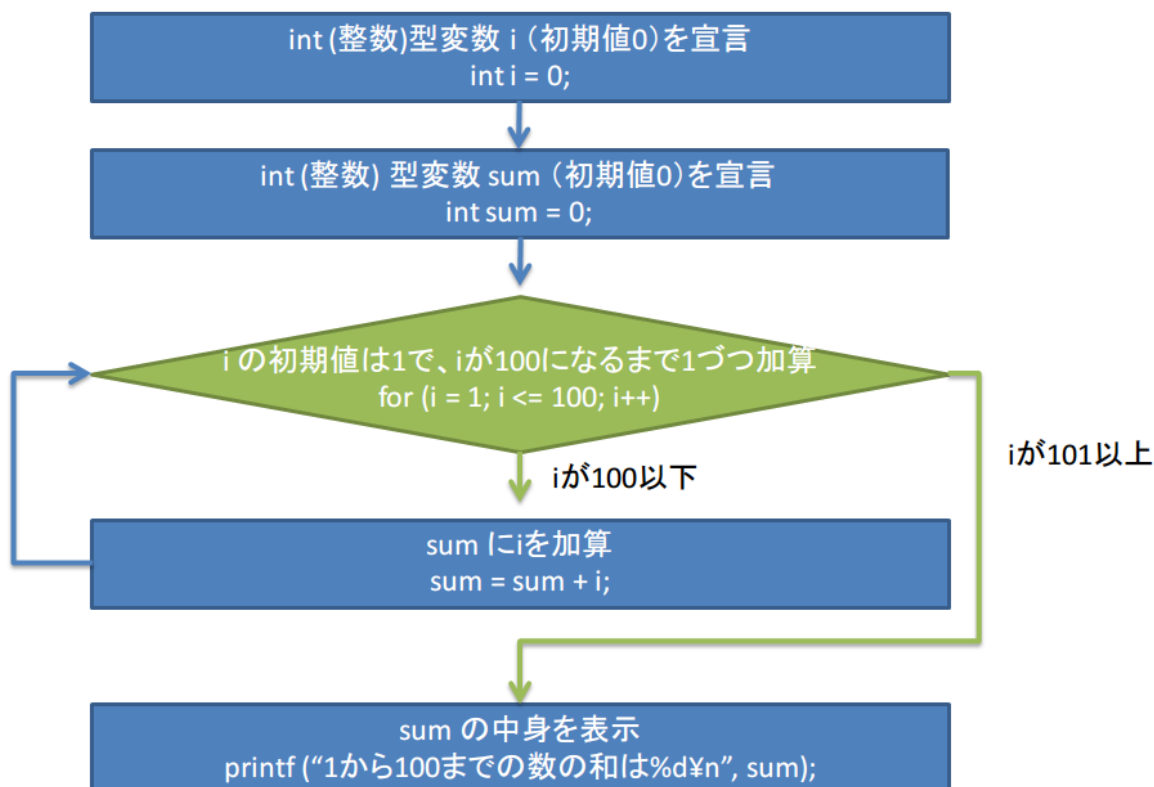
注意 †

1. for文, while文などの繰り返し構造を用いること。

†

ヒント †

for文を用いた場合の処理のフローチャート



レポート †

提出先

<https://www.kagoshima-u.ac.jp/manaba/>
の「プログラミング言語I及び演習」のレポートより提出すること

期限

演習当日の23:55まで

内容

Microsoft Wordを用いてレポートを作成すること。
レポートファイル(Wordファイル)には、ソースコードやキャプチャした実行画面などを貼り付けて、manabaのレポートから提出すること。

下記について記述すること。

演習1.1

(ビルド前のソース, 最初のビルドで見つかったエラーの一覧, エラー除去後のソース, 実行結果)

課題1.1

(ビルド前のソース, 最初のビルドで見つかったエラーの一覧, エラー除去後のソース, 実行結果)

課題1.2

(ビルド前のソース, 最初のビルドで見つかったエラーの一覧, エラー除去後のソース, 実行結果)

結論

(今回の演習を通じて学んだことについて客観的な事実を書く。主観的な内容は所感に書くこと)

相談

(誰かと相談した場合、**相手の名前**、**相談内容**、**所要時間**を書くこと。)

所感

(感想や意見などを自由に書いてよい。)

テンプレート

[Word表紙](#)

Last-modified: 2022-04-12 (火) 13:40:27 (363d)

Site admin: [anonymous](#)

PukiWiki 1.5.3 © 2001-2020 [PukiWiki Development Team](#). Powered by PHP 7.4.3. HTML convert time: 0.006 sec.