

Programming Basis

Project Interception 第1回講習会

プログラミングを始める前に・・・

- 「目的」や「目標」を持って行動したことはありますか？
- どんな些細なこと、不純な動機も「目的」「目標」です。
- プログラミングでは、これを明確化させることが

とても重要です

頭の体操です

- ちょっとしたワークをやりましょう。
- 各チーム3人に、別々の内容の紙をお配りしています。
- 中身は誰にも見せないでください。

頭を使います

- スライドでは、次のような例を考えます。みなさんは、プリントの例で考えてください。

高速バスの予約

- 料金は次のように決めます。

月	基本料金
7月	800円
8月	1300円
9月	900円

出発時間	時間料金
5時～	200円
6時～	0円
7時～	100円

番号	乗降地点
1	三島駅北口
2	静岡駅北口
3	バスタ新宿

区間	距離料金
1↔2	300円
1↔3	1000円
2↔3	1500円

- 料金は、これらのうち、客の仕様にあったものを選んだときの和です。

目的・目標を明確化する

- お題から、最終的にどうなるべきか考えます。
- => 高速バスの料金を計算し、客に提示する。

手順を言語化する

- お題は、表と簡単な説明しかありません。
- 表から料金を計算するために必要な情報を考えます。
- 出発する月、出発時間、行先、出発地点
 1. 出発する月を聞く。
 2. 出発時間を聞く。
 3. 行先を聞く。
 4. 出発地点を聞く。

手順を言語化する

- 基本料金を計算します。
5. 出発する月が7月のとき、基本料金は800円
 6. 出発する月が8月のとき、基本料金は1300円
 7. 出発する月が9月のとき、基本料金は900円

月	基本料金
7月	800円
8月	1300円
9月	900円

手順を言語化する

- 時間料金を計算します。
8. 出発時間が5時のとき、時間料金は200円
 9. 出発時間が6時のとき、時間料金は0円
 10. 出発時間が7時のとき、時間料金は100円

出発時間	時間料金
5時～	200円
6時～	0円
7時～	100円

手順を言語化する

- 距離料金を計算します。

11.行先が1かつ到着地点が2のとき、距離料金は300円

12.行先が2かつ到着地点が1のとき、距離料金は300円

13.行先が1かつ到着地点が3のとき、距離料金は1000円

14.行先が3かつ到着地点が1のとき、距離料金は1000円

15.行先が2かつ到着地点が3のとき、距離料金は1500円

16.行先が3かつ到着地点が2のとき、距離料金は1500円

17.それ以外は処理を中止する

区間	距離料金
1←>2	300円
1←>3	1000円
2←>3	1500円

手順を言語化する

- すべて足し合わせます。

18.基本料金＋時間料金＋距離料金を表示します。

手順を言語化する

- みなさんもやってみましょう！シンキング／タイム・・・
- 出来上がったら、2枚目の太枠で囲ってある部分に、その手順通りに計算したらどのような結果が得られるか確認してみましょう。
- バスの例において、「基本料金・時間料金」にあたるような数値や文字列も書いてください。途中でどんな値が生成されたかを確認しましょう。

グループワークです

- 各自書いてもらったプリントをグループの中で時計回りに交換してください。
- プリントに書いてある手順通りに計算して、2枚目のプリントに、結果を書いてください。
- 書く内容は、さきほど計算したときと同じような感じです。
- 終わったら、もう一度時計回りに回して、同じように計算してください。

グループワークです

- Aから順番に、全員の計算結果をくらべてください。
- 全員揃っていますか？
- 揃っていれば、その手順書は素晴らしいということです。
- 揃っていなければ、まだ詰めが甘いところがあるということです。
- (単純に読み間違えているかもしれないが・・・)

解釈が揃うように手順書を書き換えることを

デバッグといいます

プログラミング言語に近づける

- このままでは、コンピュータにやさしくないので、プログラミングの5要素を中心に、プログラミング言語に近づけます。

大事なこと

- 大事なことを先に。
- みなさん、文章を書くとき、「句読点」をつけますか？
- プログラムでは、後ろにセミコロンをつけます。
- セミコロンをつけないのは、句読点がない文章を書いているのと同じです。

大事なこと

- セミコロンがないソースコードは、コンピュータからすると次のように見えています。

```
printf("あなたの学籍番号（学科）を入力してください。\\n")
```

 ダイオードといえば次の4点を思い出してくださいア

先日点呼当番が事前にチェックを入れて点呼表を提出していたということがありました、今度は外泊者が3人いたのにも関わらず、棟長が点呼報告書に0と記入していました、要はせっかく点呼当番、階長が点呼をしっかり行っているのに、点呼表を見て数の転記ミスをチェックをし数えて記入する棟長の役目さえできていないということで、ダブルチェックが全く機能していない重大なミスです、特外者も外泊する連絡をしていなかったようですが、それをさせるのは棟長の役目です、棟長と階長は一心同体であり、それぞれの役目と責任を果たす必要があります、聞くところによると階長を見たことがないという棟階があるそうですが、これはどういうことなのか、棟長の顔さえも分からないでは棟運営はうまくいくわけがないです、点呼・清掃・ミーティング・挨拶や履物を揃える指導など、先頭に立って指示や注意、称賛すること、率先して寮則違反やルール、マナー違反しないことが重要です

```
float weight
scanf("%f",&weight)
printf("あなたの身長は%f[cm]です。\\n",height)
printf("あなたの体重は%f[kg]です。\\n",weight)
return 1
```

えよりこのときの電圧をフエラー電圧といいVZ であらわしますVF 同様VZ もほぼ一定ですシリコンダイオードといえばVF=0.7[V]ですダイオードにはダイオードを製造する時点で決まる値と、回路の中で変わる値があります変わらない 順方向電圧降下 VF短絡電流 Isツエナ

プログラミングの5要素

1. 情報を保持する(var)
2. もし(if)
3. N回繰り返す(for)
4. ~ときずっと繰り返す(while)
5. グループ化(void)

プログラムの8割はこれで
できている！！！！

1. 情報を保持する(var)

- 「変数」といいます。使う前に、以下のようにして「使う宣言」します。下のように、複数の値を保持できるものを「配列」といいます。

種類 名前 = 値;

種類 名前[] = {値0, 値1, 値2, ..., 値n};

1. 小数 基本料金 = 500;
2. 小数[] 倍率 = {0.5, 0.9, 0.8, 0.7};
3. 基本料金 = 基本料金 × 倍率[1];
4. 基本料金を表示する;

結果

3. 基本料金 = 450;
4. 「450」と表示される;

1. 情報を保持する(var)

- 種類には次のようなものがあります。
- 整数、小数、文字(1文字)、文字列、真偽
- いかなる変数も、途中で値を変更できます。ただし、今回は、種類が文字列のものは途中で値を変更できません。(後述)

2. もし～～なら(if)

- 「条件分岐」といいます。条件が真のとき、実行されます。わかりにくければ、ふりがなを振るとよいです。

```
1. 整数 距離料金=0;  
2. If (行き先==1 && 到着地点=2){  
3.     距離料金=300;  
4. }else if (行き先==1 && 到着地点==3){  
5.     距離料金=1000;  
6. }else{  
7.     プログラム終わり;  
8. }
```

もし

```
If (条件){  
    ～やること～
```

そうでないとき、もし

```
}else if (条件){  
    ～やること～
```

そうでないとき

```
}else{  
    ～やること～  
}
```

2. もし～～なら(if)

- 条件にもルールがあります。

条件	書き方	備考
AとBが等しい	<code>A==B</code>	数値／文字何でもok
AとBが等しくない	<code>A!=B</code>	数値／文字何でもok
AよりBのほうが大きい	<code>A<B</code>	数値のみ
AよりBのほうが小さい	<code>A>B</code>	数値のみ
AはB以上	<code>A>=B</code>	数値のみ
AはB以下	<code>A<=B</code>	数値のみ
AとBが等しい かつ AとCが等しい	<code>A==B && A==C</code>	両方成り立たてば真 それ以外は偽
AとBが等しい または AとCが等しい	<code>A==B A==C</code>	両方成り立つ場合も真

3. N回繰り返す(for)

- 決まった回数だけ繰り返します。繰り返す量は、変数を指定できます。ただし、種類は整数のみです。

繰り返す

```
for N回{ 今はi回目;  
    ～やること～  
}
```

1. 整数 繰り返し数=0;
2. 客に繰り返し数を聞く;
3. for (繰り返し数)回{ 今はi回目;
4. 50Xiを表示する;
5. }

結果

2. 繰り返し数は？ << 3と入力
4. (i=1) 50
4. (i=2) 100
4. (i=3) 150

4. ～のときずっと繰り返す(while)

- 条件が真のとき、ずっと繰り返します。条件の前に!があると、条件が偽のとき、ずっと繰り返します。

ずっと

```
while (条件){  
    ～やること～  
}
```

1. 整数 数字=0;
2. while (数字<=10){
3. 客に数字を入力させる;
4. }

結果

3. 数字は？ << 3と入力
 2. (数字=3<=10が真)
 3. 数字は？ << 15と入力
 2. (数字=15<=10が偽)
- 終了

5. グループ化(void)

- 「関数」といいます。同じ処理・複雑な処理をまとめることです。

グループ化

```
void 名前(種類 A, 種類 B,...){  
    ～処理の内容～  
    return;  
}
```

グループ化

```
種類 名前(種類 A, 種類 B,...){  
    ～処理の内容～  
    return データ;  
}
```

5. グループ化(void)

```
1. 小数 抵抗値(小数 電圧, 小数 電流){
2.     return 電圧/電流;
3. }
4. void 表示(文字列 名前, 小数 抵抗値){
5.     「<名前> : <抵抗値>Ω」 と表示;
6.     return;
7. }
8. 小数 電圧=0;
9. 電圧を聞く;
10. 小数 抵抗=抵抗値(電圧, 5);
11. 表示("R1", 抵抗);
```

結果

```
9. 電圧は？ <<10
10. (関数「抵抗値」)
2. (10/5=2を返す)
10. (抵抗=2)
11. (関数「表示」)
5. R1 : 2Ω
6. (戻る)
終了
```

プログラミングの5要素

- Discord E-Learning Systemの例です。
- ソースコードファイルは計917ファイルあります。
 1. 情報を保持する(var)：2053件
 2. もし(if)：373件
 3. N回繰り返す(for)：189件
 4. ～ときずっと繰り返す(while)：18件
 5. グループ化(void)：559件

手順を5要素で書き直す

1. 出発する月を聞く。
2. 出発時間を聞く。
3. 行先を聞く。
4. 出発地点を聞く。

1. 整数 出発する月=0;
2. 整数 出発時間=0;
3. 整数 行先=0;
4. 整数 出発地点=0;
5. 出発する月を聞く;
6. 出発時間を聞く;
7. 行先を聞く;
8. 出発地点を聞く;

手順を5要素で書き直す

5. 出発する月が7月のとき、
基本料金は800円
6. 出発する月が8月のとき、
基本料金は1300円
7. 出発する月が9月のとき、
基本料金は900円

```
9. 整数 基本料金=0;  
10.if (出発する月==7){  
11.     基本料金=800;  
12.}else if (出発する月==8){  
13.     基本料金=1300;  
14.}else if (出発する月==9){  
15.     基本料金=900;  
16.}
```

手順を5要素で書き直す

- みなさん、各自のプリントでやってみましょう！！

パソコンを起動してください

- ここからは、C言語で記述していきます。

日本語の動詞にコードを与える

日本語	C言語
表示する・伝える	print(内容)
聞く	scanf(内容, &変数名)
加算	+
減算	-
乗算	*
除算	/
整数	int
小数	float
文字列	char 名前[]="";

文字列以外の、整数や小数という種類を持つ変数の値は、いつでも変更できます。

しかし、C言語の仕様上、種類が文字列の変数の値は変更できません。

print/scanf構文

- プログラミング基礎演習でやったと思いますが . . .
- printで、変数の中身を表示させたいとき
 - `printf("でーた : %d,%f", data,shosu);`
- scanfで、キーボード入力を受けたいとき
 - `scanf("%d", data);`
- のように記述します。
- やさしいC 対応ページ : p38,p66

英語の辞書を持っていますか

- C言語では変数名に日本語は使いません。
- 適切な英語を見つけて、書き直してください。

1. 整数 出発する月=0;

2. 整数 出発時間=0;

3. 整数 行先=0;

4. 整数 出発地点=0;

5. 出発する月を聞く;

6. 出発時間を聞く;

7. 行先を聞く;

8. 出発地点を聞く;

1. int departureMonth=0;

2. int departureHour=0;

3. int destination=0;

4. int leavePoint =0;

5. scanf("%d", departureMonth);

6. scanf("%d", departureHour);

7. scanf("%d", destination);

8. scanf("%d", leavePoint);

C言語に変換する

```
9. 整数 基本料金=0;
10.if (出発する月==7){
11.    基本料金=800;
12.}else if (出発する月==8){
13.    基本料金=1300;
14.}else if (出発する月==9){
15.    基本料金=900;
16.}
```

```
9. int baseCharge=0;
10.if (departureMonth==7){
11.    baseCharge=800;
12.}else if (departureMonth==8){
13.    baseCharge=1300;
14.}else if (departureMonth==9){
15.    baseCharge=900;
16.}
```

C言語に変換する

3. N回繰り返す(for)の書き方がかなり違うので、記載しておきます。

```
1. 整数 繰り返し数=0;  
2. 客に繰り返し数を聞く;  
3. for (繰り返し数)回{ 今はi回目;  
4.     50Xiを表示する;  
5. }
```

```
1. int repeats=0;  
2. scanf("%d", &repeats);  
3. for (int i;i<repeats;i++){  
4.     printf("%d\n",50*i);  
5. }
```


C言語に変換する

このように、各行に対して1対1で対応します。
C言語のおまじないを掲載しておきます。

```
#Include <stdio.h>
//関数の定義はここに書く
int main(void){
    //処理はここに書く
}
```

それでは書いて、コンパイル&実行しましょう！

次回の内容は・・・

- 超音波センサが届くので、それを使った超音波レーダを解説します。
- Arduinoキットを持ってきてください。