

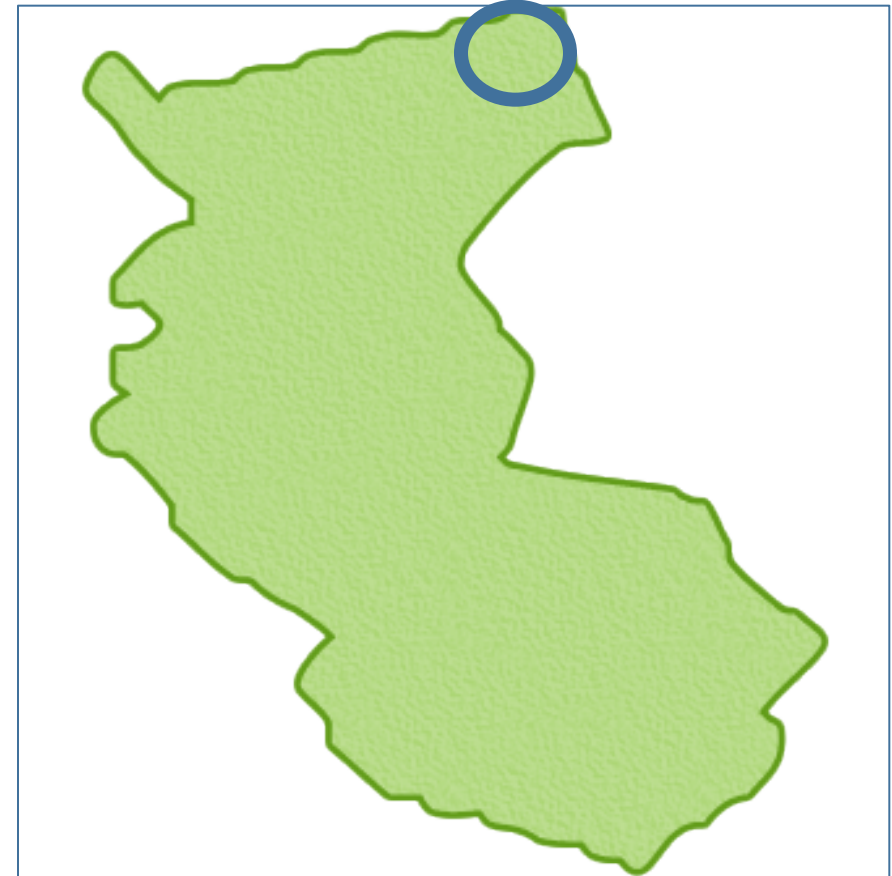
作成演習を通して自ら学ぶプ ログラミング教育

和歌山県立紀北工業高等学校

北山 浩司

紀北工業高等学校について

- 和歌山の北部橋本市
- 大阪(なんば)まで電車で50分
(和歌山で一番大阪に近い)
- 工業高校(機械科、電気科、システム化学科)
- 1学年4クラス(定員160人)、
男子約130人、女子約30人



情報関連科目への取り組み

- 情報は代替科目(情報技術基礎)
- システム化学科では、実習でも情報関係
- 選択科目で、プログラミング技術、ハードウェア技術なども設定

プログラミングについて

- プログラミングはすべての学科で1年次に
- ここ3, 4年はすべてC言語(それ以前は学科によってFullBasic, C)
- 1年次では、順次処理、分岐、繰り返しまで



簡単な内容でも定着は難しい

プログラミング学習での問題点(難点)

1. 暗記がむずかしい
2. 暗記しても使えない(作れない)
3. そもそも何をしているのがわかりにくい
4. 実習をしても、作業になる

など

プログラミング学習での問題点1

1. 暗記が難しい

- プログラミング学習では、直接思考と関係の無い部分が多くある。
- 英単語が、特殊な綴り、通常使わない単語、がある。

プログラム例

```
#include<stdio.h>
int main(void)
{
    int n,a;
    printf("数を入力");
    scanf("%d",&n);
    a=n%2;
    if(a==0)
        printf("偶数¥n");
    else {
        printf("奇数¥n");
    }
    return 0;
}
```

ほぼ呪文

謎の単語

謎の文法

ほぼ呪文

プログラミング学習での問題点2

2. 暗記しても使えない(作れない)

生徒はとりあえず暗記



単語 (printf, scanf, if, else, for などなど) は覚える。



使い方がわからない？



```
if("数を入力");  
scanf(&n);  
n%2=a;  
for(a==0) {  
    include ("偶数¥n");  
}
```


プログラミング学習での問題点3

3. そもそも何をしているのがわかりにくい

• 作れる

- [カレーの作り方]
- 1) 野菜、肉などをきざむ。
- 2) 鍋で煮込む
- 3) カレーをとり皿に盛り食卓へ

• 作れない

- [底辺、高さを入力、三角形の面積の出力]
- 1) 底辺、高さをキーボードから入力
- 2) 公式より面積を計算
- 3) 面積をとり出力画面へ

プログラミング学習での問題点4

4. 実習をしても、作業になる

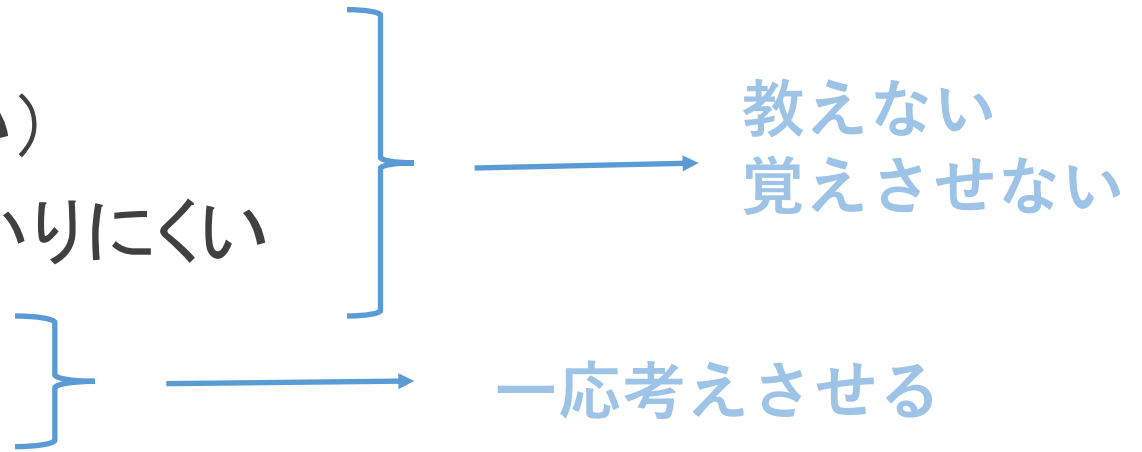
●プログラムを写すだけ

- ・良いプログラム例なら実行をたのしむ
- ・エラーはすべてチェックする→理解するより探し出すほうが簡単
- ・キーボード入力が楽しい

●PCを触りたいだけ

提案

1. 暗記がむずかしい
2. 暗記しても使えない(作れない)
3. そもそも何をしているのがわかりにくい
4. 実習をしても、作業になる



実習の構成

- 簡単な説明 (5, 6行のみ)
- 入力プログラム例
(ソースリスト提示)
- 改変課題
(文章で提示)

[説明]

- `scanf` … 入力する。
この命令があるとキーボードからの入力を待ちます。
- 利用例 `scanf("%d",&a)`
変数 `a` に入力する。`a` の前に `&` を付けること。

問題3

1) 次のプログラムは整数を `a`、`b` の入力し、和、差を出力するものである。このプログラムを入力し、実行しなさい。(プロジェクト名 03wasa)

[プログラム]

```
#include<stdio.h>
int main(void)
{
    int a,b,wa,sa;
    scanf("%d",&a);
    scanf("%d",&b);
    wa = a + b;
    sa = a - b;
    printf("wa=%d\n",wa);
    printf("sa=%d\n",sa);
    return 0;
}
```

キーボードから `a` に入力
キーボードから `b` に入力

2) 上記のプログラムで積と商 (整数でよい) を表示するように変更しなさい。なお、変数名を `seki`、`syou` に変更すること。

実習の構成

1. 学習者は「簡単な説明」を聞く(読む)。
2. ソースプログラムを入力して実行を行う。
3. 改変課題では入力のなかで中身を推理(基本は教えない)する。
4. 修正して実行を行う。
5. エラーが出たり、思い通りの結果が出ないときは、それを見てさらに推理する。

基本はコンピュータとの対話の中で自ら考える演習とする。

教材例

- テーマ: 入力処理についての演習
- ・和と差についての処理について入力。
- ・実行(エラーはこのときに)。
- ・計算式の中に「+」と「-」を意識できれば、そこを「*」、「/」に変更すると可能となることに学習者に気付かせる。
- ・実行するとメッセージに「wa=」、「sa=」と表示されるので、そのときにメッセージのwa、saをseki、syouniに変更することに気付くと、変数名も変更すべきことに気付くことになる。

教材内容(簡単な説明)

- [説明]
- scanf ... 入力する。
- この命令があるとキーボードからの入力を待ちます。
- 利用例 `scanf("%d",&a)`
- 変数aに入力する。aの前に&を付けること。

教材内容(サンプルプログラム)

```
• #include<stdio.h>
• int main(void)
• {
•     int a,b,wa,sa;
•     scanf("%d",&a);           キーボードからaに入力
•     scanf("%d",&b);           キーボードからbに入力
•     wa = a + b;
•     sa = a - b;
•     printf("wa=%d¥n",wa);
•     printf("sa=%d¥n",sa);
•     return 0;
• }
```


教材内容(変更課題)

- 2) 上記のプログラムで積と商(整数でよい)を表示するように変更しなさい。なお、変数名をseki、syounに変更すること。

課題一覧

総合問題(※)では単独での文章問題を数問用意して復習と設定している。ただし、学習者が気付けばそれまでのどれかの問題を改造すれば実現できる問題ばかりである。

問題 1	1 行出力
問題 2	変数定義、足し算
問題 3	入力処理
問題 4	分岐処理 (if文)
問題 5	実数
問題 6	分岐処理 余り計算
問題 7	総合問題 (※)
問題 8	繰り返し処理 (1 ~ 1 0 0)
問題 9	繰り返し処理 (1 ~ N)
問題 1 0	繰り返し処理 (入力の合計、平均)
問題 1 1	繰り返し処理 (素数判定)
問題 1 2	繰り返し処理 (while文)
問題 1 3	配列
問題 1 4	最大値
問題 1 5	総合問題 (※)

演習の展開

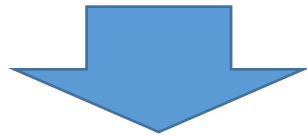
- 最初操作方法を示した(ひとつプログラムを入力、実行する場面を提示)あと、自由に演習を行なわせる。
- 理解の良い生徒は自らどんどん演習を進めさせる。
- わからなくなる生徒にはそのときに個別に指導をサポート。
- 基本的には生徒が自主的に行うように努めるようにする。

評価

- ・生徒・・・比較的興味を持ってせられる。
自分のペースでできるので時間をもてあまさない。
その問題のみが解ける。(文法の説明がないので)
- ・教師・・・基本的な操作を教えれば、生徒が勝手にすすめる。
(細かく教えないので)質問が多くなる。

実習環境について

- 問題点
 - 言語のシステムはインストールが難しい
 - 生徒はファイル概念がわからない(作ったソースがどこかにいってしまう)
 - システムが重い



インストールしない(オンラインで)
ファイル保存しない

Ideone.com(サイト)を



The screenshot shows the Ideone.com interface. At the top, the site name "ideone.com" is visible on the left, and a "new" button with a house icon is on the right. Below the header, there are action buttons: "edit" (pencil icon), "fork" (fork icon), "download" (download icon), and "copy" (copy icon). The main area contains a C program with the following code:

```
1. #include <stdio.h>
2.
3. int main(void) {
4.     int i,sum;
5.     sum=0;
6.     for(i=1;i<=100;i++) {
7.         sum=sum+i;
8.     }
9.     printf("sum=%d",sum);
10.    return 0;
11. }
```

Below the code, a status bar indicates "Success #stdin #stdout 0s 9424KB" and "comments (0)". Underneath, there are two sections for input and output:

stdin (copy icon): Standard input is empty

stdout (copy icon): sum=5050

Ideone.comの特徴

長所

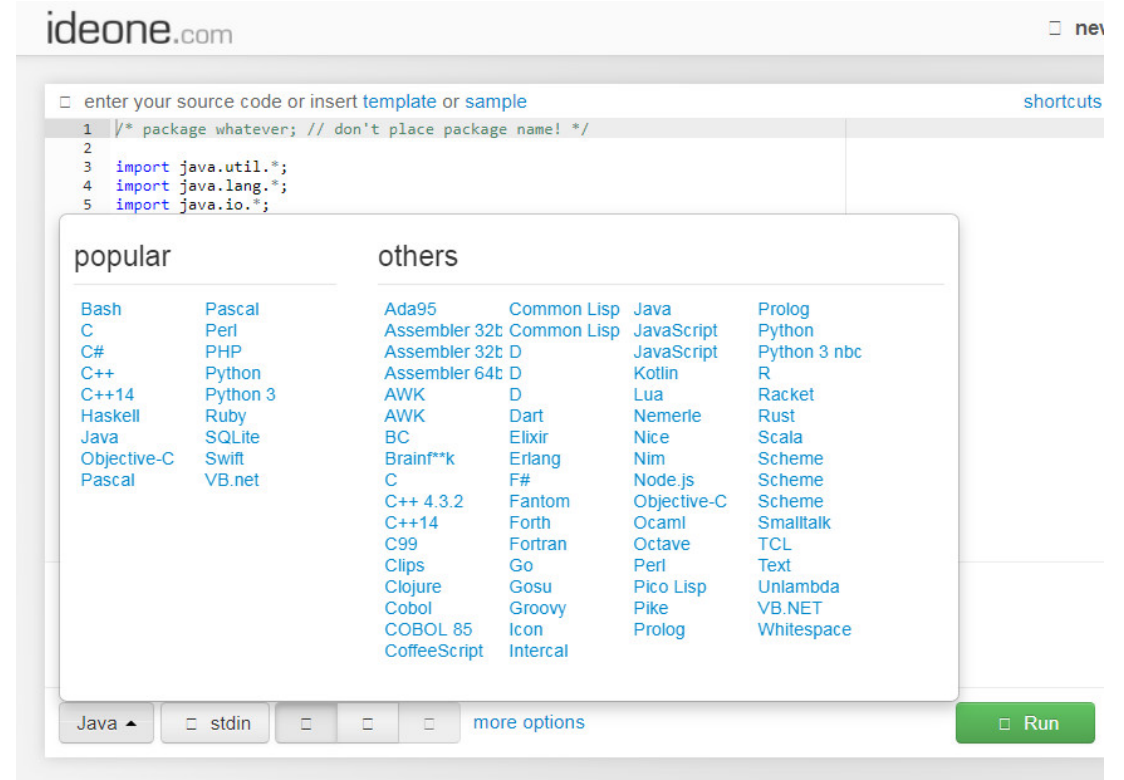
- ・インストール不要
- ・どのPCからでもすぐに演習が可能
- ・「呪文(includeなど)」の部分はあらかじめ用意される

短所

- ・実行時間が長い
- ・入カデータは実行前にあらかじめその領域(stdin、input)に置いておかないといけない。
- ・一定時間放置するとエラーが出る。

Ideone.comの特徴2

- 利用できる言語が多種
- C , C# , Java , Pascal , Python , Ruby , Swift
- をはじめ色々



(追記) クラブ活動に(コンピュータ部)

さらに高度なプログラミング教育を



授業では時間と生徒のモチベーションが足りない



クラブ活動で実現

クラブ(コンピュータ部)について

- プログラミングを基本練習に
- プログラミングコンテスト(情報オリンピック、パソコン甲子園、スーパーコンピューティングコンテスト)に出場、またその対策
- 週に5日、6日。運動部並み(それ以上の)練習
- プログラムが重要な資格取得対策(基本情報など)

クラブ(コンピュータ部)実績

大会実績

- ・日本情報オリンピック
11年連続本選出場中
- ・パソコン甲子園プログラミング部門
16回中7回本選出場
- ・スーパーコンピューティングコンテスト
10年間で8回本選出場

など

進学実績(AOなど特徴入試)

- ・筑波大学情報学群
- ・立命館大学2名
- ・関西大学4名
- ・長岡科学技術大学

など

最後に

- ・プログラミングコンテストの上位はほとんど普通科進学校。
- ・プログラミングの能力を持つ生徒は今でも高く評価。
- ・学力が伴うことでさらに効果。
- ・将来社会のプログラミングに対して社会の認知度があがればさらに期待？
- ・まずは授業「情報」から？