情報Ⅰ　プログラミングのエッセンス提出課題

下の文章を読み、どんブロックでプログラムを作成して実行せよ。

実践1　合計の求め方＜テキストP.12-13 ＞

■ 選択課題１０　　毎日の貯金額の合計

あなたは貯金をしています。1 日目に 100 円、2 日目に 200 円、3 日目に 300 円のように、毎日、前日より 100 円多く貯金することにしました。

1 日目から 10 日目までに貯金した金額の合計を求め、表示するプログラムを作成しなさい。

■使用する変数

- goukei：貯金額の合計
- i：日数を数えるための変数

■期待される実行結果

5500

■ 選択課題１１　　合格者の得点の合計

6 人の生徒のテストの得点が、配列 Tokuten に格納されています。

Tokuten = [68, 75, 92, 54, 70, 83]

70 点以上を合格とします。配列を先頭から順に調べ、合格者の得点だけを加算して、合計点を表示するプログラムを作成しなさい。

ただし、配列の添字は 0 から始まるものとする。

■使用する変数

- Tokuten：生徒 6 人の得点を格納する配列
- goukei：合格者の得点の合計
- i：配列の添字

■期待される実行結果　　320

■ チャレンジ課題４　センサーの異常値を探せ

測定データの中に、おかしな値が混じっていたら見つけられるでしょうか？

研究室で 10 回測定した温度が、配列 Ondo に格納されています。

Ondo = [22, 24, 31, 19, 27, 28, 35, 23, 26, 18]

20 度以上 30 度未満を正常な測定値とします。正常な値の合計、個数、平均値を求めます。

次の条件を満たすプログラムを作成しなさい。

1. 配列を先頭から順番に調べる。
2. 20 度以上 30 度未満の測定値だけを合計する。
3. 条件を満たす測定値の個数を数える。
4. 合計値と個数から平均値を求める。
5. 合計値、個数、平均値をそれぞれ表示する。

配列の添字は 0 から始まるものとする。また、正常な測定値が 1 つもない場合は、平均値を計算せず、そのことが分かるように表示すること。

■期待される実行結果

合計：150 度　個数：6 回　平均：25 度

実践 2　個数・日数・人数の数え方（テキスト P.14〜15）

下の文章を読み、どんブロックでプログラムを作成して実行せよ。

■ 選択課題 12　暑い日を数えよう

ある 1 週間の最高気温が、配列 Kion に格納されています。

Kion = [24, 28, 31, 22, 27, 30, 26]

最高気温が 28 度以上となった日数を求め、表示するプログラムを作成しなさい。

配列の添字は 0 から始まるものとする。

■使用する変数：Kion（最高気温の配列）、nissuu（日数）、i（配列の添字）

■期待される実行結果：3

■ チャレンジ課題５　ラッキーデーは何日ある？

ゲームのアイテムをたくさん獲得できる日は、何日あるのでしょうか？

ゲームで 10 日間に獲得したアイテムの個数が、配列 Uriage に格納されています。

Uriage = [12, 7, 15, 9, 18, 6, 14, 10, 5, 16]

1 日に 10 個以上獲得できた日を「ラッキーデー」とします。ラッキーデーの日数(nissu)と、10 日間に占める割合(wariai)を求めて表示しなさい。

割合は、日数 ÷ 10 × 100 で求めること。使用する変数は自分で考えて設定すること。

■期待される実行結果：nissu：6　wariai：60

実践 3　最大・最小の求め方（テキスト P.16〜17）

下の文章を読み、どんブロックでプログラムを作成して実行せよ。

■ 選択課題 13　いちばん売れた月を調べよう

ある店の 1 年間の月別売上額（単位：百万円）が、配列 Uriage に格納されています。

Uriage = [12, 15, 9, 18, 14, 21, 17, 16, 20, 13, 19, 11]

1 年間で最も売上額が大きかった値を求め、表示するプログラムを作成しなさい。添字は 0 から始まるものとする。

■使用する変数：Uriage（月別売上額の配列）、saikou（最大売上額）、i（配列の添字）。

■期待される実行結果： 21

■ チャレンジ課題 6　モンスターが最も多く現れる日は？

冒険者ギルドが、月曜日から日曜日までのモンスター出現数を記録しました。

Monsutaa = [250, 270, 300, 230, 220, 240, 290]

Youbi = ["月", "火", "水", "木", "金", "土", "日"]

モンスターが最も多く出現した曜日と数、最も少なかった曜日と数を求め、両方を表示するプログラムを作成しなさい。配列の添字は 0 から始まるものとする。

使用する変数は自分で考えて設定すること。

■期待される実行結果：MAX：水 300 体　MIN：金 220 体

実践 4「割り算の余りの利用」（テキスト P.18～19）

素数の判定

■ 選択課題 14　入力した整数が素数か判定しよう

外部から入力された 2 以上の整数について、素数かどうかを判定し、「素数です」または「素数ではありません」と表示するプログラムを作成しなさい。

■使用する変数：seisuu（判定する整数）、wareta（割り切れる数があったかを記録する変数）、
　　i（割る数）

■ 条件：：2 から seisuu - 1 までの整数で割り切れるかを調べること。

■ チャレンジ課題 7　素数だけが開ける暗号箱

素数には、どんな秘密が隠されているのでしょうか？秘密の箱を開ける暗証番号候補が 6 個あります。ただし、箱を開けられるのは素数だけです。

Seisuu = [17, 21, 29, 1, 31, 35]

配列の中から素数(sosu)だけを表示し、最後に素数がいくつあったか(kosuu)を表示するプログラムを作成しなさい。1 は素数ではないものとする。

```
sosu 17
sosu 29
sosu 31
kosuu 3
```

それぞれの数について、2 からその数の 1 つ前までの整数で割り切れるかを調べること。

実践 5「関数の定義」（テキスト P.20〜21）

引数と戻り値を使おう

■ 選択課題 15　税込み価格を求めよう

商品の税抜き価格が変数 Nedan に格納されています。税率を 10%として、税込み価格を戻り値とする関数 Zeikomigaku(kingaku) を定義し、その関数を使って税込み価格を表示するプログラムを DNCL で作成しなさい。

> 税抜き価格：Nedan = 1000
> 関数の仕様：Zeikomigaku(kingaku) は、引数 kingaku の税込み価格（税抜き価格の 1.1 倍）を戻り値とする。

■使用する変数：Nedan（税抜き価格）、zeikomi（関数から受け取った税込み価格）

■ 条件：：関数を定義し、Nedan を引数として関数を呼び出すこと。

正しい実行結果

```
1100
```

■ チャレンジ課題 8　閉店直前！お買い物ミッション

関数を使えば、買い物の計算をまとめられるでしょうか？4 つの商品価格が配列 Nedan に入っています。関数 割引後価格(kingaku) は、引数として価格を受け取り、10%引き後の価格を戻り値とします。関数は次のように定義済みとします。

```
関数定義 割引後価格(kingaku)：
|   戻り値(kingaku * 0.9)
```

配列の各商品の割引後価格を求め、その合計金額(goukei)を表示するプログラムを作成しなさい。関数の呼び出しには、配列の要素を引数として渡すこと。

Nedan = [500, 800, 1200, 1500]

■正しい実行結果：Goukei 3600

実践 6「探索と並べ替え」（テキスト P.22〜23）

配列の■ 条件：検索とバブルソート

■ 選択課題 16　在庫が少ない商品を見つけよう

店舗で販売する商品名が配列 Shouhin、在庫数が配列 Zaiko に格納されています。添字が同じ要素は同じ商品を表します。在庫数が 5 個以下の商品について、商品名と在庫数を表示するプログラムを DNCL で作成しなさい。

```
Shouhin = ["ノート", "ペン", "消しゴム", "定規", "ファイル"]
Zaiko = [12, 4, 3, 8, 5]
```

■使用する変数：Shouhin（商品名の配列）、Zaiko（在庫数の配列）、i（配列の添字）

■ 条件：在庫数が 5 個以下の商品だけを表示する。配列の添字は 0 から始まるものとする。

■表示例：ペン 4、消しゴム 3、ファイル 5

■ チャレンジ課題 9　並べ替えが苦手なデータはある？

隣り合う数字を交換していくだけで、きれいな順番に並べ替えられるのでしょうか？5 人のプレイヤーのスコアが配列 Kazu に入っています。バブルソートを使って、数値を小さい順（昇順）に並べ替え、隣り合う要素を交換した回数も表示するプログラムを作成しなさい。

Kazu = [12, 5, 18, 9, 14]

①隣り合う要素を左から順に比較する。②左の値が右の値より大きいとき、2 つの値を交換する。③交換した回数を数える。④最後に、並べ替え後の配列と交換回数を表示する。

■実行結果

```
並べ替え後 [5, 9, 12, 14, 18]
交換回数 4
```

実践 7 予測可能なシミュレーション（テキスト P.24〜25）

■ 選択課題 17　毎日のポイントの増加を調べよう

ゲームで、最初に 350 ポイントを持っています。1 日ごとに 120 ポイントずつ増えるとき、7 日間のポイントの推移を表示するプログラムを DNCL で作成しなさい。各日のポイントは、増加した後の合計を表示すること。

> 初期値：point = 350
> 1 日ごとの増加：120 ポイント
> ■使用する変数：point（ポイントの合計）、day（日数）
> ■ 条件：：day を使った繰り返しで、1 日目から 7 日目まで表示する。

■正しい実行結果

> 1: 470　2: 590　　...7: 1190

■ チャレンジ課題 10　コラッツ予想に挑戦

どんな正の整数でも、決められた規則を繰り返すと最後には 1 になるのでしょうか？整数 n を次の規則で変化させます。偶数なら 2 で割り、奇数なら 3 倍して 1 を足します。n が 1 になるまで繰り返し、途中の数、1 になるまでの回数(kaisu)、途中で現れた最大値(saikou)を表示するプログラムを作成しなさい。

初期値：n = 7　使用する変数：n（現在の数）、kaisu（繰り返した回数）、saikou（これまでの最大値）。最初の数を変えて、結果を比べてみよう。

実行例（初期値 n = 7 の場合）

7 → 22 → 11 → 34 → 17 → 52 → 26 → 13 → 40 → 20 → 10 → 5 → 16 → 8 → 4 → 2 → 1

回数：16　最大値：52

実践 8 乱数を使ったシミュレーション　(p.26-27)

■ 選択課題 18　くじのボーナスを判定しよう

ゲームで、1 回の抽選ごとに 1 以上 10 以下の整数をランダムに発生させます。出た数が 1 または 2 なら「ボーナス」、それ以外なら「通常」と表示するプログラムを DNCL で作成しなさい。出た数も表示すること。

使用する関数：整数乱数(a, b) は、a 以上 b 以下のランダムな整数を返す。
■使用する変数：me（発生させた整数）
■ 条件：乱数は 1 回だけ発生させ、出た数に応じて判定結果を表示する。

正しい実行結果（me = 2 の場合）

```
2
ボーナス
```

■ チャレンジ課題 11　モンティ・ホール問題に挑戦

ドアを選び直すだけで、当たりやすさが変わるのでしょうか？3 つのドアのうち 1 つに当たりがあります。最初に 1 つを選んだ後、司会者は残りのドアから、当たりではないドアを必ず 1 つ開けます。その後、最初の選択を変えた場合と変えなかった場合について、**10, 000 回の実験で当たりになった回数と割合を求めるプログラム**を作成しなさい。各試行では、当たりのドア(atari)と最初に選ぶドア(sentaku)を整数乱数(1, 3)で発生させること。最初の選択を変えなかった場合は、sentaku と atari が同じなら成功とする。選び直した場合は、sentaku と atari が異なれば成功とする。3 つのドアでは、司会者が必ず外れのドアを 1 つ開けた後、残ったもう 1 枚に選び直すと、この判定と同じ結果になる。司会者が開けるドアそのものを乱数で決める必要はありません。

■使用する変数：atari（当たりのドア）、sentaku（最初に選ぶドア）、kae（選び直した場合の当たり回数）、kaena（選び直さない場合の当たり回数）、i（繰り返し用）

■条件：各試行で当たりのドアと最初に選ぶドアをそれぞれ 1 回ずつ発生させ、選び直す場合と選び直さない場合の当たり回数を数える。最後に、それぞれの回数を 10,000 で割って割合を求める。

■実行結果の目安：選び直す場合は約 2/3、選び直さない場合は約 1/3（乱数のため、実行ごとに結果は変わる）。

＜解答例＞DNCL 言語を参考にして、どんブロックで実行してみよう

選択課題 10　毎日の貯金額の合計

```
goukei ← 0


1 から 10 まで 1 ずつ増やしながら繰り返す
    goukei ← goukei + i * 100
を繰り返す


goukei を表示する
```

選択課題 11　合格者の得点の合計

```
Tokuten ← [68, 75, 92, 54, 70, 83]
goukei ← 0


i を 0 から 5 まで 1 ずつ増やしながら繰り返す
    もし Tokuten[i] が 70 以上 ならば
        goukei ← goukei + Tokuten[i]
    を実行する
を繰り返す


goukei を表示する
```

チャレンジ課題 4　センサーの異常値を探せ

```
Ondo ← [22, 24, 31, 19, 27, 28, 35, 23, 26, 18]
goukei ← 0
kosuu ← 0


i を 0 から 9 まで 1 ずつ増やしながら繰り返す
    もし Ondo[i] が 20 以上 かつ Ondo[i] が 30 未満 ならば
        goukei ← goukei + Ondo[i]
        kosuu ← kosuu + 1
    を実行する
を繰り返す


goukei を表示する
kosuu を表示する
もし kosuu が 0 より大きい ならば
    heikin ← goukei / kosuu
    heikin を表示する
そうでなければ
    「正常な測定値はありません」を表示する
を実行する
```

選択課題 12　暑い日を数えよう

```
Kion ← [24, 28, 31, 22, 27, 30, 26]
nissuu ← 0


i を 0 から 6 まで 1 ずつ増やしながら繰り返す
    もし Kion[i] が 28 以上 ならば
```

```
        nissuu ← nissuu + 1
    を実行する
を繰り返す


nissuu を表示する
```

チャレンジ課題５　ラッキーデーは何日ある？

```
Uriage ← [12, 7, 15, 9, 18, 6, 14, 10, 5, 16]
nissu ← 0


i を 0 から 9 まで 1 ずつ増やしながら繰り返す
    もし Uriage[i] が 10 以上 ならば
        nissu ← nissu + 1
    を実行する
を繰り返す


wariai ← nissu / 10 * 100
nissu を表示する
wariai を表示する
```

選択課題 13　いちばん売れた月を調べよう

```
Uriage ← [12, 15, 9, 18, 14, 21, 17, 16, 20, 13, 19, 11]
saikou ← Uriage[0]


i を 1 から 11 まで 1 ずつ増やしながら繰り返す
    もし Uriage[i] が saikou より大きい ならば
        saikou ← Uriage[i]
    を実行する
を繰り返す


saikou を表示する
```

チャレンジ課題６　モンスターが最も多く現れる日は？

```
Monsutaa ← [250, 270, 300, 230, 220, 240, 290]
Youbi ← [「月」, 「火」, 「水」, 「木」, 「金」, 「土」, 「日」]
max ← Monsutaa[0]
min ← Monsutaa[0]
maxYoubi ← Youbi[0]
minYoubi ← Youbi[0]


i を 1 から 6 まで 1 ずつ増やしながら繰り返す
    もし Monsutaa[i] が max より大きい ならば
        max ← Monsutaa[i]
        maxYoubi ← Youbi[i]
    を実行する
    もし Monsutaa[i] が min より小さい ならば
        min ← Monsutaa[i]
        minYoubi ← Youbi[i]
    を実行する
を繰り返す


「MAX：」と maxYoubi と max を表示する
「MIN：」と minYoubi と min を表示する
```

選択課題 14　入力した整数が素数か判定しよう

```
seisuu ← input()
wareta ← 0


i を 2 から seisuu - 1 まで 1 ずつ増やしながら繰り返す
    もし seisuu を i で割った余り が 0 ならば
        wareta ← 1
    を実行する
を繰り返す


もし wareta が 0 ならば
    「素数です」を表示する
そうでなければ
    「素数ではありません」を表示する
を実行する
```

チャレンジ課題 7　素数だけが開ける暗号箱

```
Seisuu ← [17, 21, 29, 1, 31, 35]
kosuu ← 0


i を 0 から 5 まで 1 ずつ増やしながら繰り返す
    sosu ← 1
    もし Seisuu[i] が 2 未満 ならば
        sosu ← 0
    を実行する
    j を 2 から Seisuu[i] - 1 まで 1 ずつ増やしながら繰り返す
        もし Seisuu[i] を j で割った余り が 0 ならば
            sosu ← 0
        を実行する
    を繰り返す
    もし sosu が 1 ならば
        Seisuu[i] を表示する
        kosuu ← kosuu + 1
    を実行する
を繰り返す


kosuu を表示する
```

選択課題 15　税込み価格を求めよう

```
関数 Zeikomigaku(kingaku) を
    戻り値 ← kingaku * 1.1
    戻り値を返す
とする


Nedan ← 1000
zeikomi ← Zeikomigaku(Nedan)
zeikomi を表示する
```

チャレンジ課題 8　閉店直前！お買い物ミッション

```
関数 割引後価格(kingaku) を
    戻り値 ← kingaku * 0.9
    戻り値を返す
とする
```

```
Nedan ← [500, 800, 1200, 1500]
goukei ← 0

i を 0 から 3 まで 1 ずつ増やしながら繰り返す
    goukei ← goukei + 割引後価格(Nedan[i])
を繰り返す

goukei を表示する
```

選択課題 16　在庫が少ない商品を見つけよう

```
Shouhin ← [「ノート」, 「ペン」, 「消しゴム」, 「定規」, 「ファイル」]
Zaiko ← [12, 4, 3, 8, 5]

i を 0 から 4 まで 1 ずつ増やしながら繰り返す
    もし Zaiko[i] が 5 以下 ならば
        Shouhin[i] と Zaiko[i] を表示する
    を実行する
を繰り返す
```

チャレンジ課題 9　並べ替えが苦手なデータはある？

```
Kazu ← [12, 5, 18, 9, 14]
koushinkaisu ← 0

n を 4 から 1 まで 1 ずつ減らしながら繰り返す
    i を 0 から n - 1 まで 1 ずつ増やしながら繰り返す
        もし Kazu[i] が Kazu[i + 1] より大きい ならば
            kari ← Kazu[i]
            Kazu[i] ← Kazu[i + 1]
            Kazu[i + 1] ← kari
            koushinkaisu ← koushinkaisu + 1
        を実行する
    を繰り返す
を繰り返す

Kazu を表示する
koushinkaisu を表示する
```

選択課題 17　毎日のポイントの増加を調べよう

```
point ← 350

day を 1 から 7 まで 1 ずつ増やしながら繰り返す
    point ← point + 120
    day と point を表示する
を繰り返す
```

チャレンジ課題 10　コラッツ予想に挑戦

```
n ← 7
kaisu ← 0
saikou ← n

n を表示する
```

```
n が 1 でない 間、繰り返す
    もし n を 2 で割った余り が 0 ならば
        n ← n / 2
    そうでなければ
        n ← n * 3 + 1
    を実行する
    kaisu ← kaisu + 1
    もし n が saikou より大きい ならば
        saikou ← n
    を実行する
    n を表示する
を繰り返す


kaisu を表示する
saikou を表示する
```

選択課題 18　くじのボーナスを判定しよう

```
me ← 整数乱数(1, 10)
me を表示する


もし me が 1 または me が 2 ならば
    「ボーナス」を表示する
そうでなければ
    「通常」を表示する
を実行する
```

チャレンジ課題 11　モンティ・ホール問題に挑戦

```
kae ← 0
kaena ← 0


i を 1 から 10000 まで 1 ずつ増やしながら繰り返す
    atari ← 整数乱数(1, 3)
    sentaku ← 整数乱数(1, 3)
    もし sentaku が atari ならば
        kaena ← kaena + 1
    そうでなければ
        kae ← kae + 1
    を実行する
を繰り返す


wariaiKae ← kae / 10000
wariaiKaena ← kaena / 10000
kae と wariaiKae を表示する
kaena と wariaiKaena を表示する
```