**コンピュータの動作/アセンブラプログラミング ワークシート**

1. **開発環境**

|  |  |
| --- | --- |
| CASLシミュレーター | 検索 |



[Web版ＣＡＳＬⅡシミュレータ](https://www.chiba-fjb.ac.jp/fjb_labo/casl/casl2.cgi)　を指定する。

1. **アセンブラ言語仕様**

|  |  |  |  |
| --- | --- | --- | --- |
| 命令 | 一般形式 | 意味 | 例 |
| 数値を入れる | LAD rg1,数値 | 数値を指定したrg1に入れる | LAD GR1,7 |
| 加算 | ADDA rg1,rg2 | rg1にrgを加える | ADDA GR1, GR2 |
| ADDA rg1,adress  | 指定したアドレスの内容をrg1に加える | ADDA GR1, DATA1 |
| 減算 | SUBA rg1,g2 | rg1からrg2を引く | SUBA GR0, GR2 |
| SUBA rg1,adress | 指定したアドレスの内容をrg1から引く | SUBA GR0, 7 |
| メモリ→レジスタ | LA rg1,adress | 指定したアドレスの内容を指rg1に入れる | LA GR1, DATA1 |
| レジスタ→メモリ | ST rg1,adress | rg1の内容を指定したアドレスに入れる | ST GR1, 7 |
| 比較する | CPA rg1,rg2 | 指定した2つの数値比較する。もし等しい場合はZeroフラグが1, それ以外は0 | CPA GR1, GR2 |
| CPA rg1,adress | CPA GR1, DATA1 |
| 等しい場合ジャンプ | JZE adress | 等しい場合(Zeroフラグが1)の時指定したアドレスから実行 | ST GR1, 7 |
| 無条件にジャンプ | JUMP adress 　 | 無条件に指定したアドレスから実行 | JUMP L1 |
| 終了 | RET | プログラムの終了を示す | RET |
| プログラム開始 | STRAT | アセンブラの開始を指定 | STRAT |
| プログラム最後 | END | アセンブラの終了を指定 | END |
| メモリへの数値の確保 | DC 数値 | メモリを確保し、値をいれてやく | DC 3 |

1. **課題**

打ち込み/動作確認

1. 課題01 : 単純な加算(レジスタ使用)

左のプログラムを打ち込んで実行して、レジスタの内容を確認してください。

K1 START

 LAD GR1,1

 LAD GR2,2

 ADDA GR1,GR2

 RET

 END

打ち込み/動作確認

1. 課題02 : 単純な加算(メモリ使用)

K2 START

 LD GR1,DATA1

 ADDA GR1,DATA2

 ST GR1,DATA3

 RET

DATA1 DC 3

DATA2 DC 4

DATA3 DC 0

 END

左のプログラムを打ち込んで実行して、レジスタ・メモリの内容を確認してください。

打ち込み/開発

1. 課題03 : 比較

K3 START

 LD GR1,DATA1

 CPA GR1,DATA2

 JZE HITOSHI

 JUMP STOP

HITOSHI LAD GR2,1

 ST GR2,KEKKA

STOP RET

DATA1 DC 3

DATA2 DC 3

KEKKA DC 0

 END

左のプログラムを打ち込んで実行して、レジスタ・メモリの内容を確認してください。

プログラムのDATA1とDATA2の内容を修正したメモリKEKKAに0のままになるようにしてください。

1. 課題04 : ???

打ち込み/動作確認

K4 START

 LAD GR0,0

 LD GR1,DATA2

 LAD GR2,1

 LAD GR4,0

LSTART CPA GR1,GR4

 JZE LEND

 ADDA GR0,DATA1

 SUBA GR1,GR2

 JUMP LSTART

LEND ST GR0,KEKKA

 RET

DATA1 DC 3

DATA2 DC 4

KEKKA DC 0

 END

左のプログラムを打ち込んで実行して、レジスタ・メモリの内容を確認してください。

何をしているプログラムか考えてみてください。

開発

1. 課題05 : 1～10までの合計

繰り返しを使って、1～10までの合計を計算するプログラムを作ってください。