

工業科学習指導案

令和2年10月 第2学年 指導者 原 裕哉

1 単元名 「配列とポインタ」

2 学習指導要領上の位置付け

プログラミング技法については、プログラム言語の規則の習得に偏ることなく、アルゴリズムの基本構造である順次、選択、繰り返しの構造などの論理的思考の学習を重視し、実際にコンピュータを活用し実習や演習を通して、具体的に理解させる。

ア 順次型のプログラム

言語特有のプログラムの書き方、データの型、入出力、計算、変数の使い方やプログラムの実行順とコンピュータの動作原理などについて理解させること。

イ 選択型のプログラム

変数、オブジェクトなどの状態、割り込み発生などの動作によって異なった処理を行う事象を取り上げ、適切な選択条件の設定方法、多分岐選択、複数条件分岐の場合の選択条件によるプログラムの作成方法について理解させること。

ウ 繰り返し型プログラム

反復処理の事象を取り上げ、適切な繰り返しの条件の設定方法や、繰り返し範囲内部の値や状態の変化について体験的に理解させること。

エ プログラムの標準化

プログラムの実行順を調べることなどを通して、プログラムのトレース手法や開発支援を効率的に行うサービスプログラムの操作について体験的に理解させること。

3 目標

配列とポインタに関する実習や演習を通して、次の事項を身に付けることができるよう指導する。

ア プログラムを作成しようとする意欲があり、正しく動作するプログラムを完成させる粘り強い態度を身に付けるようにする。（関心・意欲・態度）

イ 文法、定数や変数の種類や使用方法、入出力の方法、分岐や繰り返しなどプログラムの実行制御方法などの実際にプログラムを作成するための方法を考えるとともに、与えられた処理を行うための手順を分析し、プログラムとして表現する実践的な能力を身に付けるようにする。

（思考・判断・表現）

ウ 流れ図からソースプログラムを作成し、コンパイラを用いて実行可能プログラムを作成できるようにする。プログラムを検査することができ、目的どおりに処理を行っているかを確認し、バグがあればそれを取り除くことができるようにする。（技能）

エ 定数や変数の種類や使用方法、入出力の方法、プログラムの実行制御の方法など文法を理解し、それを基に実際にプログラムを作成する知識を身に付けるようにする。（知識・理解）

4 指導計画 ※別紙参照

5 本時の展開 (10/16 時間)

(1) ねらい

アドレス演算子 (&) と間接演算参照子 (*) について、メモリ内部の模式図を用いて考える活動を通して、&と*演算子を使ったプログラムのメモリ内部の動きを説明できるようにする。

(2) 展開

学習活動 ・予想される生徒の反応	時間	○指導上の留意点 ◎研究上の手立て 評価項目<方法(観点)>
<p>1 本時の目標を設定し、ポインタについて復習をする。</p> <p>・本時の目標や進め方が分かった。</p>	10分	<p>○前時の復習や本時の目標や進め方の提示を工夫し、本時の学習への期待感や意欲を高められるようにする。</p>
<p>[目標] アドレス演算子 (&) と間接演算参照子 (*) によって変数に入る値はどのようなものになるのか考えよう。</p>		
<p>2 本時の課題とワークシートやグループワークの進め方を聞き、以下の課題に取り組む。</p> <p>(1) 提示されたプログラムにおけるデータの変化をワークシートに記入する。(個人)</p> <p>・プログラムを読み取りながらその時の変数の状態をワークシートに記入する。</p> <p>(2) ラミネートシートを使って自分の考えを伝える。(グループワーク)</p> <p>・グループでの話し合い活動が深まらない。</p> <p>(3) 班で考えた結果をクラス全体に発表する。</p>	<p>5分</p> <p>5分</p> <p>10分</p> <p>10分</p>	<p>◎ワークシートと、グループごとにラミネートシートを配付し、メモリ内部のデータの変化を可視化して考える活動を通して、データの変化を理解する。</p> <p>○自ら考え、相手に説明することを考えながらワークシートを進めていくことを指示する。</p> <p>◎グループワークにより教え合うことで理解度を高める。</p> <p>○グループワークによる交流が活発でない班にはヒントを出して話し合い活動が行われるようにする。</p> <p>○プログラム中のある時点におけるデータの様子の写真を撮っておくよう指示する(データが変更されてしまうため)。</p> <p>◎他者の発表を聞くことで理解度を高める。</p> <p>○発表者の説明が滞ったら、班員が助言するよう促す。</p>
<p>アドレス演算子 (&) と間接演算参照子 (*) の働きについて理解し、データがどのように変化しているか説明することができる。<ワークシート・発言(2)></p>		
<p>3 学習をまとめる。</p> <p>・スライドでの本時の振り返りをした後、Google Forms を利用した振り返り小テストを行い、提出する。</p>	10分	<p>○本時の学習で分かったことを生徒に問い掛け、生徒とやり取りしながら、まとめを行う。</p>

6 板書計画

10/16

目標

アドレス演算子 (&) と間接演算参照子 (*) によって変数に入る値はどのようなものになるのか考えよう。

プロジェクタ

- ① 前時までの復習
- ② ワークシートの説明
- ③ グループワークの進め方の説明
- ④ 本時のまとめ

指導計画 情報技術科 第2学年 単元名「配列とポインタ」(全16時間計画)

目標	<p>配列とポインタに関する実習や演習を通して、次の事項を身に付けることができるよう指導する。</p> <p>ア プログラムを作成しようとする意欲があり、正しく動作するプログラムを完成させる粘り強い態度を身に付けるようにする。(関心・意欲・態度)</p> <p>イ 文法、定数や変数の種類や使用方法、入出力の方法、分岐や繰り返しなどプログラムの実行制御方法などの実際にプログラムを作成するための方法を考えるとともに、与えられた処理を行うための手順を分析し、プログラムとして表現する実践的な能力を身に付けるようにする。(思考・判断・表現)</p> <p>ウ 流れ図からソースプログラムを作成し、コンパイラを用いて実行可能プログラムを作成できるようにする。プログラムを検査することができ、目的どおりに処理を行っているかを確認し、バグがあればそれを取り除くことができるようにする。(技能)</p> <p>エ 定数や変数の種類や使用方法、入出力の方法、プログラムの実行制御の方法など文法を理解し、それを基に実際にプログラムを作成する知識を身に付けるようにする。(知識・理解)</p>			
評価規準	<p>(1) プログラムを作成しようとする意欲があり、正しく動作するプログラムを完成させる粘り強い態度を身に付けている。(関心・意欲・態度)</p> <p>(2) 多量のデータを効率よく扱うために、配列やポインタを用いたプログラムを書くことができ、またそのプログラムの意味を説明することができる。(思考・判断・表現)</p> <p>(3) 問題を解決するためのアルゴリズムを考え、それを基にプログラミングすることができ、バグがあれば取り除くことができる。(技能)</p> <p>(4) 配列やポインタを使うための文法を理解し、それを基に実際にプログラムを作成する知識を身に付けている。(知識・理解)</p>			
過程	時間	○ねらい 目標	・振り返り (意識)	評価項目 〈方法 (観点)〉
つかむ	1	<p>○一つの変数に複数のデータを格納するにはどのようにすればよいか、メモリ内部の模式図を用いて考える活動を通して、配列の概念を理解できるようにする。</p> <p>配列はどのようなデータ構造になるか模式図を用いて表現できるようにする。</p>	<p>・配列を使えば一つの変数に複数のデータを扱えることは分かった。どのように記述すれば、配列を用いたプログラムが書けるのだろうか。</p>	<p>・配列とは何かデータ構造を模式図として表そうとしている。〈ノート・発言(1)〉</p>
追究する	1	<p>○配列の模式図を基に、配列のサイズと要素番号について考える活動を通して、配列のサイズと要素番号の添え字の関係を説明できるようにする。</p> <p>配列のサイズと要素番号の関係はどのような決まりがあるか表現できるようにする。</p>	<p>・配列のサイズは格納できるデータの個数を表し、要素番号の添え字は0から始まるということが分かった。</p>	<p>・配列の宣言やサイズや添え字について模式的に考え、説明することができる。〈ノート・発言(2)〉</p>
	1	<p>○配列に文字列を代入するのはどのようにすればよいかをプログラミング演習を通して文字型配列と文字列の関係を理解する。</p> <p>配列に文字列を代入するプログラムを作成できるようにする。</p>	<p>・文字型配列の場合は char 型で配列を宣言する必要があることが分かった。</p>	<p>・文字型配列の宣言や初期化の方法についてプログラミングすることができる。〈成果物・ノート(3)〉</p>
	1	<p>○Null 文字の意味とその働きについてプログラミング演習を通して理解する。</p> <p>文字列の終わりは終端記号 (Null 文字) が入ることを理解できるようにする。</p>	<p>・文字型配列の場合、文字列の終わりに終端記号である Null 文字が入ることが分かった。そのため、宣言する要素数は1バイト余分に宣言する必要があることが分かった。</p>	<p>・文字列の終端には Null 文字が入り、文字列は配列の先頭から Null 文字までを文字列であることを理解している。〈ノート・発言(4)〉</p>
	1	<p>○文字列関数について、プログラミング演習を通して、文字の複写や文字列の長さを求めたりすることができるようにする。</p> <p>文字列関数にはどのような種類があり、どのような働きをするか理解できるようにする。</p>	<p>・文字列関数という関数があることが分かった。これを使えば文字データを処理する際、簡潔で分かりやすいプログラムが書けることが分かった。</p>	<p>・文字列関数を使って文字の複写や長さを求めるプログラムを作成することができる。〈成果物・ノート・発言(3)〉</p>
	3	<p>○多次元配列の概念について理解するため、模式図にデータを入れて考える活動を通して、多次元配列を使ったプログラムを書くことができるようにする。</p> <p>多次元配列を用いたプログラムを作成できるようにする。</p>	<p>・多次元配列を使うことで2次元や3次的にデータを管理、計算することができ、データを効率よく処理することが分かった。</p>	<p>・多次元配列の概念を模式的に表し、説明することができる。多次元配列を使ったプログラムを書くことができる。〈発言・ノート・成果物(3)〉</p>

	<p>1 ○メモリ上のアドレスを扱うポインタ変数について、メモリ内部の模式図を用いて考える活動を通して、ポインタの概念を理解する。</p> <p>ポインタ変数に入るデータはどのようなデータが入るか模式図を用いて表現できるようにする。</p>	<ul style="list-style-type: none"> ポインタ型とはメモリ上のアドレスを扱う型であり、ポインタ変数にはアドレスが代入されないということが分かった。 	<ul style="list-style-type: none"> ポインタ型変数とはどのような変数であるか、模式図を用いて説明することができる。〈発言・ノート(1)〉
	<p>2 ○アドレス演算子(&)と間接演算参照子(*)について、メモリ内部の模式図を用いて考える活動を通して、&と*演算子の使ったプログラムのメモリ内部の動きを説明できるようにする。</p> <p>アドレス演算子(&)と間接演算参照子(*)によって変数に入る値はどのようなものになるか模式図を用いて表現できるようにする。</p>	<ul style="list-style-type: none"> ポインタ型変数を使用するために必要な&と*の使い方が分かって、ポインタ変数を使ったプログラムを作ることができた。 	<ul style="list-style-type: none"> アドレス演算子と間接参照演算子はどのようなものかを説明することができ、ポインタ変数を使ったプログラムを書くことができる。〈発言・ノート・成果物(2)〉
	<p>1 ○ポインタ変数を使った配列内部の演算について、メモリ内部の模式図を用いて考える活動を通して、配列を用いた場合とポインタを用いた場合とで、同じことができることを理解する。</p> <p>配列の演算をする際、配列を用いた場合と、ポインタを用いた場合の違いについて説明できるようにする。</p>	<ul style="list-style-type: none"> ポインタ変数を用いても配列内の合計や平均を求めることができた。ポインタを使うことで配列名を指定しなくても、間接的にデータを操ることができることが分かった。 	<ul style="list-style-type: none"> 配列を用いた場合とポインタを用いた場合を比較して、同じことができることを説明することができる。〈発言・ノート・成果物(2)〉
	<p>2 ○ポインタと文字列について、メモリ内部の模式図とプログラムを作成する活動を通して、ポインタを用いることですっきりとした分かりやすいプログラムが書けることを理解できるようにする。</p> <p>文字列を操作する際、ポインタを使ったプログラムを作成できるようにする。</p>	<ul style="list-style-type: none"> 文字列を使うプログラムにポインタを用いることで分かりやすいプログラムが書けることが分かった。 	<ul style="list-style-type: none"> ポインタを使ったプログラムから、メモリ内部の様子について模式図を用いて説明することができ、ポインタを用いると分かりやすいプログラムが書けることを理解している。〈発言・ノート・成果物(4)〉
まとめ	<p>2 ○演習問題を解く活動を通して、学習内容を定着できるようにする。</p> <p>既習事項を用いて、プログラムを作成することができ、また、バグがあれば取り除くことができるようにする。</p>	<ul style="list-style-type: none"> ポインタを使ったプログラムでも、配列を使ったプログラムとして書くことができる事が分かった。配列とポインタは密接な関係があることが分かった。 	<ul style="list-style-type: none"> 単元の基本的な考え方を理解しており、プログラムを作成できる技能を身に付けている。〈ノート・発言・成果物(3)〉