

学びのデザインシート（本時）

主体的・対話的で深い学びを実現する授業構想【工業／実習】

1. 対象（生徒の実態の概要）

年間を通して9テーマの実習を受ける。基礎学力に大きな差はないが、授業への集中力や理解度、参加意識には差を感じる。主体性の高い生徒もおり、対話に参加し授業を盛り上げてくれる。一方、参加意識の低い生徒への働きかけに工夫を要する。授業の内容は前年度の工業技術基礎で学習してきた内容の応用編と捉えることができる。

2. 単元名 「総合実習」（全8時間）

3. 単元の目標

知識及び技能	工業に関する要素技術を総合化した内容について工業の各分野での学びを踏まえて理解するとともに、工業に携わる者として必要な技術を身に付ける。
思考力, 判断力, 表現力等	電気・電子・通信の分野に関する技術に着目して、工業の各分野に関連する個々の要素技術を総合化した技術に関する課題を見いだすとともに解決策を考え、科学的な根拠に基づき結果を検証し改善する。
学びに向かう力, 人間性等	電気・電子・通信の分野に関する要素技術を総合化した内容について自ら学び、工業の発展に主体的かつ協働的に取り組む。

4. 言語活動

エキスパート活動、ジグソー活動で個々が思考したアルゴリズムの意見交換をする。
プログラムの修正後に、修正箇所や修正方法を発表し、合理的なプログラミングの意見を出し合う。

5. 本時の目標

課題8のプログラミング学習を通して、エキスパート活動で共有したアルゴリズムをジグソー活動の中で主体性をもって発揮できる。すなわち、対話を通して学習内容の理解を深め、課題を解決することができる。

6. 授業展開

解決したい課題や問い

既存のプログラムを理解し、課題となったプログラムに修正できる。

考えるための材料

《エキスパート活動》

次の3グループに分け、材料を基に個々で思考する。その後、グループ内で対話による意見交換をする。

- ①【既存】と【修正】のプログラムを実行し、動作によるプログラムの違いを確認する。
- ②【既存】と【修正】のプログラムを文章(言語)により説明し、違いを確認する。
- ③【既存】から【修正】プログラムへの修正方法を教示しながら確認する。

想定される活動

エキスパート活動のグループによって、次のとおりの活動が想定される。

- ①【既存】と【修正】の動作を文章(言語)化し、【既存】プログラムの修正を行う。
 - ②【修正】後の動作を想像し、文章(言語)化する。【既存】プログラムの修正を行う。
 - ③【既存】プログラムを自ら修正し、動作が変更されていることを確認する。
- ①、②のグループは文章(言語)化した段階で共有の場を持つ。
③は個人ではなく、グループで活動する。

プログラムを修正する過程でアルゴリズムを理解する。

修正課題を提示した時に既知の内容をどこに応用できるか思考する。

不明な点を教員または班員に質問したり相談したりする。

対話と思考（対話を通じた協働的な問題解決のプロセス）

《エキスパート活動》

一班を2、3人とする。

個々で思考したアルゴリズムを意見交換する。班員の意見を参考に課題解決に向かう。

《ジグソー活動》

一班を2、3人とする。

エキスパート活動を班員と共有し、互いに【修正】後の動作を対話によって確認する。

互いに相違が生じた場合は協議し、解決に向かう。

【修正】後の動作が確認できた班から【修正】プログラムの修正作業（プログラミング）を行う。

修正作業中も他の班員は修正作業を見守り、時には助言を加える。

《クロストーク》

【修正】プログラムの完成後、互いの班の【修正】プログラムの動作を確認し合う。

また、【修正】プログラムの修正箇所や修正方法を発表し合う。

より合理的な【修正】プログラムを模索し、意見を出し合う。

【修正】プログラムへの付加機能を思考および記述し、意見交換する。

学習の成果（予想される生徒のあらわれ）

《エキスパート活動》

個々で思考を繰り返し、文章（言語）化する生徒がいる。

はじめから他人と相談しながら文章（言語）化する生徒もいる。

論理的な矛盾が生じないような文章構成ができるようになる。

《ジグソー活動》

理解の進んでいる生徒が【修正】プログラムの修正作業（プログラミング）を主導する。

理解の進んでいる生徒がそうでない生徒に対して修正箇所や修正方法を説明しながら、ともに理解を深める。

理解の進んでいない生徒が理解を深めることでプログラミングに対する苦手意識が薄まる。

修正作業を見守る生徒が俯瞰的な立場により合理的なプログラミングに気づく。

《クロストーク》

互いの班の【修正】プログラムの発表を聞いて、自らの班との違いに気づく。

課題解決型の学習内容に対する抵抗感が軽減される。

【修正】プログラムへの付加機能を意見交換することでプログラミングへの意識が高揚する。

グループでの対話を通して、課題解決が図られることに意義を感じる。

自他尊重の価値を醸成し、グループにおける安心感を得る。