

全国高等専門学校 第35回 プログラミングコンテスト

奈良大会 課題部門 登録番号10001

応募校に修正依頼しなかったもの

地域が特定できる・著作権違反など

その他

(1) 登録されているタイトル・サブタイトルと、p1に記載されているタイトル・サブタイトルが異なります

第35回全国高専プロコン(奈良大会)事務局
2024年05月28日 11時15分25秒





SDGs

持続可能な開発都市
～100年続くまちづくり～

課題部門
登録番号
対象者

10001
小学校 高学年

11 住み続けられる
まちづくりを



環境問題について効果的に学びたいのに

実感しにくい！

地球温暖化問題などの規模の大きな問題ほど、進行が遅いため実感しにくく問題発見が遅れる
発見が遅れることでより問題が進行してしまい悪循環

地球温暖化って本当に起こってるの??

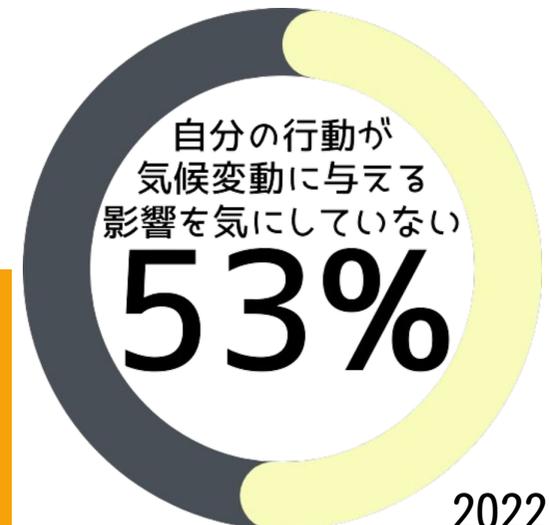
田舎で自然豊かだから問題意識が低い!!



環境問題への意識を高める必要性

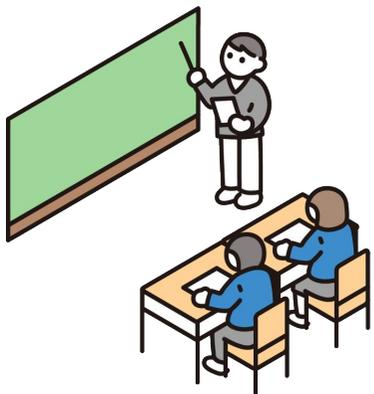
環境問題を認識しながらも自分ごとと考える人は少数
自分1人が与える影響が小さく自分から行動しない人が多数

将来を担う小学生年代から環境意識を高める必要性
→自分の住む環境への関心を高める



2022年
BCG調べ

規模の大きな環境問題の方が効果的に学びにくい



児童は授業を通して様々な環境問題に出会う

体験的な活動を十分に取り入れる必要
興味や関心を持って自分の問題として受け止めていく豊かな感受性を育む必要

環境教育指導資料[小学校編]より

しかし

実際に小学校を訪問して調査すると児童は身近な環境問題に比べて規模の大きな問題に関心を示しにくい！

身近に感じにくい

実感が湧かない

時間的視野が狭い

紙や映像での説明では分かりにくい

興味を示さないから意識が高まらない



小学校の先生

そこで大規模な環境問題の特に地球温暖化問題に着目した SDCs を提案する

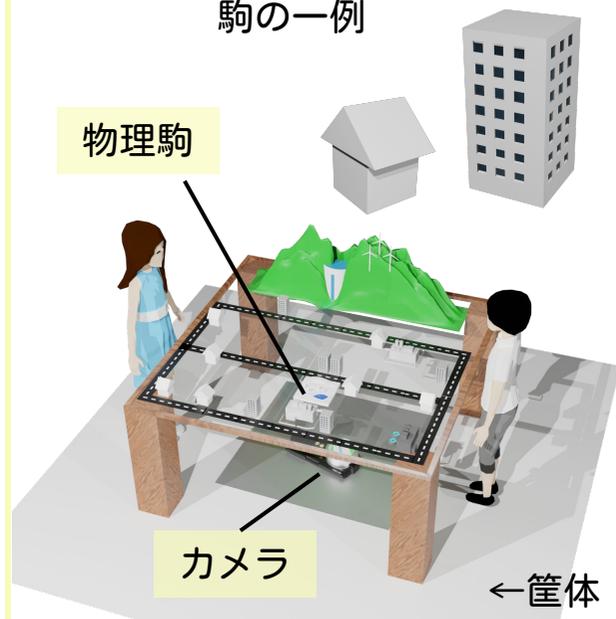
目的・効果

SDGs を通して、地球温暖化問題を触って・見て・考えて、環境問題に対する意識を高め
実生活の行動に繋げる

触る

物理ブロック駒を使って
街を形成
様々な形の駒を触って配置
→直感的な操作

駒の一例



見る

実際に配置された駒を元に
街を映像化し出力
環境状態により周囲を変化
触ったものを映像化し見る
ことで理解度向上に繋げる



考える

作った街に起こる環境問題
を通して、

- ・自分が取るべき行動
- ・起こった結果の是非
- ・今後必要な行動

について考え、環境問題に
関する意識の向上に繋げる

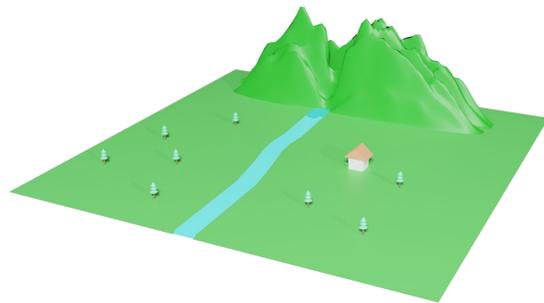


その後の実生活へ
活かす

まちづくりを進める

まずは、何もない状態から街を発展させていく
初めは街を発展させることに集中→環境問題は出現しない

まちを発展させる



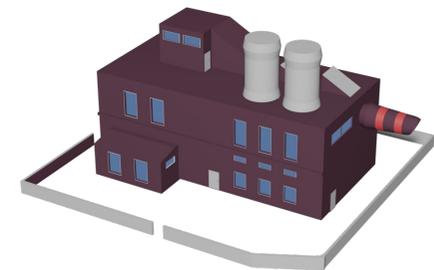
初めは家一軒から
スタート

まちづくりを進める



まちを発展させていく

生活環境に関する
需要に対応



需要に応じて必要な
ものを追加(工場など)

主体的に取り組む

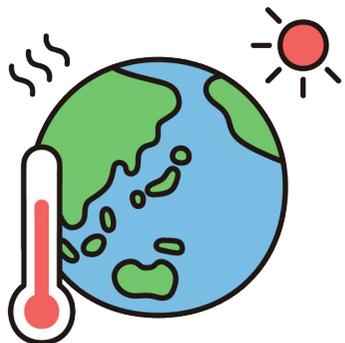
まちづくり完了後、街を操作して出現した問題を解決
→解決に最適なアイテムを自分で**考えて選び**、配置する
結果は、配置後に時間を進めるまで**わからない**
→プレイ次第でまちの存続が左右される

時間の経過

駒を配置している時は
現実世界と同じ速度

時間を進めるときは
1秒1ヶ月の速度で進む

地球温暖化問題の解決



地球温暖化による環境**問題が発生!**

100年後にまちが持続しているかどうかを決めるバロメータが変化

まちを100年以上持続させるために発生した**問題の解決**に取り組む

※持続度バロメータの詳細についてはP8を参照

問題の解決



石油発電所

二酸化炭素を多く排出
地球温暖化への影響度 **大**



風車、ソーラーパネル

持続可能なエネルギー
地球温暖化への影響度 **小**



存続年数のこり **70年**

存続年数のこり **95年**

フィードバック

結果をフィードバック



結果をもとに思考

自分のとった行動を客観視
→最善策を考える

評価の方法

環境問題を解決したか？
その問題を解決することができたか



その状態を持続可能か？
一時的な解決になっていないか

フィードバック詳細

SDCs

(いま)2124/12/31

Result

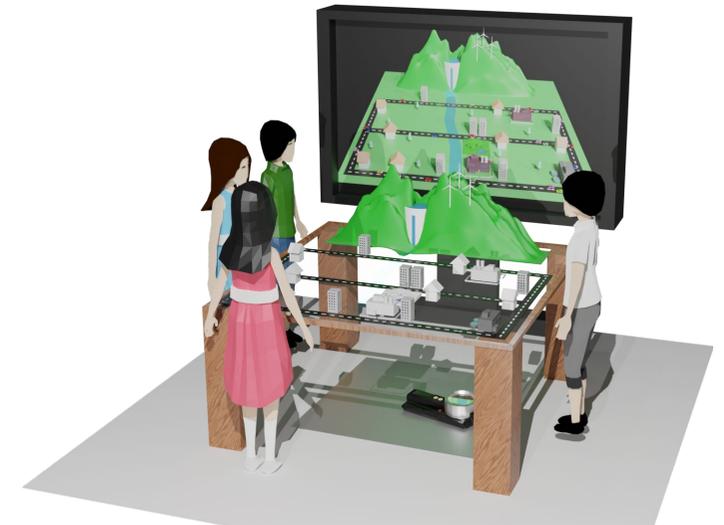
存続年数のこり **70年**



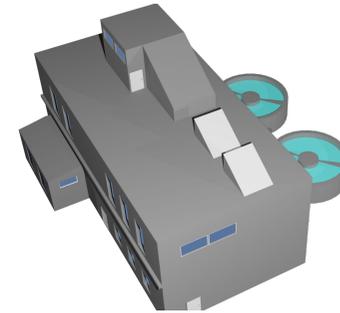
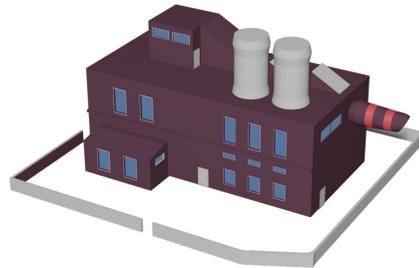
- ◎水がきれい→浄水場の設置◎
- ◎エコに発電→風力発電◎
- 空気を綺麗に→森林を増やすには何をすべき？
- △工場が減って産業が停滞→なんで工場を減らしたのかな？

終了

プレイ終了後にグループ内でもう一度考える



各パーツの二酸化炭素排出量に合わせてまちの存続バロメータを変化



計算方法：例（二酸化炭素の排出）

デンマークLCIA手法であるEDIP97のDtT法による重み付け手法を参考に式(1)を用いて現状値との乖離の程度から重み付け係数を参照した。

$$WF(j) = \frac{ER(j)_{2020}}{ER(j)_{T2030}} = \frac{1.147 \times 10^{12} [\text{kg}]}{0.760 \times 10^{12} [\text{kg}]} = 1.509 \quad (1)$$

$WF(j)$ ：影響領域 j の重み付け係数

$ER(j)_{2020}$ ：影響領域 j の参照年（2020年）における環境影響量（現状値）

$ER(j)_{T2030}$ ：影響領域 j の目標年（2030年）における環境影響量の目標値

式(1)の $ER(j)_{2020}$ がシステムプレイ中の環境影響量となる

$WF(j)$ の値が、1.0になった時100年存続する計算となる

→1.0より**大きい**場合：100年未満しか存続できない

→1.0より**小さい**場合：100年より長期に渡り存続できる

プレイ時間中に存続年数が0年となった場合、その場でまちが存続不可能となり滅びる。

二酸化炭素を減らすために
生活に必要なものを
取り除く
(発電設備をなくすなど)

存続年数が**マイナス方向**に
人々の生活を担保した上で
問題を解決する必要
 $WF(j)$ の値は二酸化炭素だけ
でなくまちが**存続できるか**
どうかで変化

大規模な環境問題は、実際に外に出て感じたりしにくい
→写真(紙ベース)や映像で伝えることが多くなってしまいう結果、身近に**感じづらい!**

SDCsでは...

実物(物理筐体)

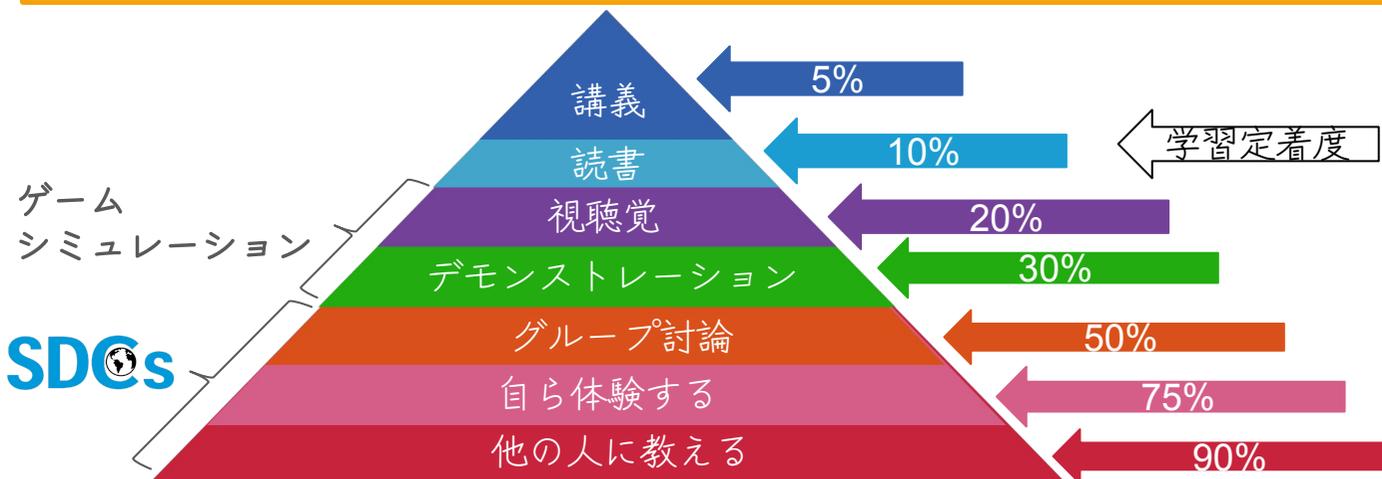
体系的に学べる
物理駒により**形のあるまち作り**が楽しめる
映像だけよりも**記憶に定着しやすい**
基本的な操作は**自分で考える**ことが求められる
環境の変化を感じる**ことが難しい**



仮想(ICT機器)

自分で作るまちの環境が目の前で
リアルタイムに変化
正確な数値をもとに**変化を早めて**表示する
体感しやすく、**理解度UP**につながる
映像だけでは**臨場感に欠ける**

実物・仮想の双方を使用することによって、**理解度・定着率を向上**させることができる



平均学習定着率が向上する「ラーニングピラミッド」



現実空間(ユーザ)

盤面に駒を配置

- 現実空間での街作りゲーム
- 駒の配置



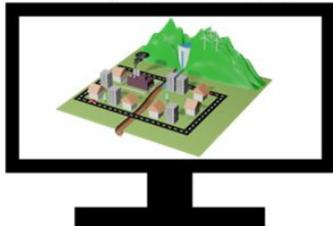
ArUcoマーカー

- 各建造物の識別
- 位置情報
- 配置状況を反映



モニターに投影

- 環境反映された街の姿
- Unityとの連携



カメラ認識

JSON形式

映像出力

仮想空間(PC)

 Python



- 画像処理
- 位置情報取得
- 位置情報の送信

 Unity



- 環境シミュレーション
- 3Dモデルの描画
- モニターへ出力
- 視点変化



類似システム

ブロックでの遊び
家や街などを作る(友達と一緒に)

シムシティ
街を発展させる
お金や経営状況を数値で理解可能



本システム

- 環境問題の度数を**数式**として、
時間変化に応じて**反映**
- **実物**→**仮想**空間反映
- 体系的に考え、学べる
- お金指標ではなく**環境指標**
- 環境問題に特化

特許調査

特開2022-9815:ゲームシステム

環境反映の有無、盤面の形状、物体検出方法において、本システムとは異なる

課題点

いかに理解度を高めるか

対象者がプレイする際の誘導方法によって、理解度や実感度が大きく変化することが予想される

そのため、効果的な誘導方法・表現の仕方をより追求したいと考えている

	4月	5月	6月	7月	8月	9月	10月
システム要件定義	<ul style="list-style-type: none"> アイデア出し 機能決定 	予選資料作成					
3Dモデルの作成		<ul style="list-style-type: none"> 建物の3Dモデルを作成 色付け 					
3Dモデルの描画			作成した3Dモデルの描画				
環境計算		<ul style="list-style-type: none"> 環境係数を用いた計算 環境問題のシミュレーション開発 					
駒の作成			3Dプリンタを使用し、建造物の駒を作成				
盤面作成			プレイする盤面の作成				
システム連携				モニタ、駒、盤面のシステムの連携			
実証投入・テスト				テスト		実証導入	

開発環境：VScode/VisualStudio/Unity/OpenCV(ArUcoMarker)/GitHub/Blender
 開発言語：Python / C#
 実行環境：Windows11 / macOS Sonoma / ウェブカメラ