

科学研究費助成事業 研究成果報告書

平成 28 年 6 月 7 日現在

機関番号：32657

研究種目：基盤研究(C) (一般)

研究期間：2013～2015

課題番号：25350275

研究課題名(和文) 地域素材を活用したCSアンプラグドの研究

研究課題名(英文) A Study on CS Unplugged utilizing Regional Materials

研究代表者

福岡 久雄 (FUKUOKA, Hisao)

東京電機大学・情報環境学部・特別専任教授

研究者番号：10370016

交付決定額(研究期間全体)：(直接経費) 2,600,000円

研究成果の概要(和文)：CS(Computer Science)アンプラグドの地域素材活用型アクティビティとして、オペレーティングシステムの主要機能であるプロセススケジューリングを学習するようなアクティビティを提案し、教材開発を行った。プロセススケジューリングのアンプラグド化には出雲神話に登場するヤマタノオロチが食事をする状況を利用した。アンプラグド化の妥当性についてはCSアンプラグドのデザインパターンによって検証した。本アクティビティを通して、学習者は具体的なスケジューリングを体験的に学習することができる。本アクティビティを高等専門学校の低学年生や大学の新生生に対して試行した結果、その有効性を確認することができた。

研究成果の概要(英文)：In this research, we have proposed a new CS Unplugged activity utilizing regional materials and have developed its teaching materials. It teaches the concept of process scheduling to its learners. In order to make the function of process scheduling unplugged, we have utilized Yamata-no-Orochi (Eight-Headed Dragon), which appears in Izumo mythology. We have verified our process of unplugging based on the CS Unplugged design pattern. This activity enables its learners to experience some concrete process scheduling methods. Through the trials of this activity with students of a technical college and a university, we have confirmed its effectiveness.

研究分野：情報工学

キーワード：CSアンプラグド 地域素材 プロセススケジューリング

1. 研究開始当初の背景

(1) CS アンプラグド (Computer Science Unplugged) とは子供向けのコンピュータ関連技術教育法の1つである。その特徴は、一見してコンピュータ関連技術とは無関係のカードゲームなどを用いることによって、「コンピュータを使うことなく」、各種コンピュータ関連技術の考え方を教授することにある。CS アンプラグドの本家ホームページや日本語教則本によれば、各種コンピュータ技術に対応して、既に 20 種類を超えるアクティビティ (学習項目) が用意されている。

(2) CS アンプラグドで扱われていない技術も数多くあることから、新たなアクティビティの開発が望まれており、事実、各所で新アクティビティの提案・開発が行われている。

2. 研究の目的

(1) 本研究では地域素材を活用した新たなアクティビティを提案・開発する。

(2) 地域素材としては、出雲神話に登場するヤマタノオロチを採用し、学習対象となるコンピュータ関連技術はオペレーティングシステム (OS) の主要機能であるプロセススケジューリングと定める。

(3) デザインパターンによる検証や複数回の試行からのフィードバックに基づいて、提案・開発するアクティビティの有効性 (教材としての成立性) を確認する。

3. 研究の方法

(1) OS の主要機能であるプロセススケジューリングをアンプラグド化する方策を検討する。

プロセスとは動作中のプログラムのことであり、OS によってその動作状態が管理される。一般に、動作中のコンピュータにおいては同時に複数のプロセスが存在し、それらが CPU 時間を奪い合うような状況にある。そこで、OS は特定のタイミングで次に実行すべき (すなわち、CPU 時間を割り当てるべき) プロセスを何らかの方針 (スケジューリングポリシー) に従って決定する。この機能をプロセススケジューリングと呼ぶ。この方針に関しては FCFS (First Come First Served, 到着順), SPTF (Shortest Processing Time First, 処理時間順), ラウンドロビン (Round-Robin) などが知られている。

プロセススケジューリングは、横取り (Preemption) のあるタイプと横取りのないタイプに大別される。横取りのあるプロセススケジューリングとは、ある特定のタイミングで OS が実行中のプロセスを中断し、他のプロセスの実行を開始することを許すようなものである。上述の FCFS や SPTF は横取りのないプロセススケジューリングであり、ラウンドロビンは横取りのあるプロセススケジューリングである。

プロセススケジューリングを、ヤマタノオ

ロチを用いてアンプラグド化するに当たって、次のような状況を設定した。

- ① 8 体のオロチが食事をするが、ある時刻に食事ができるオロチは 1 体だけである。
- ② 食糧は離散的なものであり、オロチが 1 単位時間に食べる量は 1 単位食糧とする。
- ③ 事前に何らかの方法で各オロチの食事開始時刻とそれぞれが食べる食糧量を定めておく。ここでは、学習者の意向を反映させないことがポイントである。
- ④ 食事可能なオロチは必ず食事をしようとする。

この状況では、必然的にオロチ間で食糧の奪い合いが発生する。これは、コンピュータにおいて複数のプロセスが CPU 時間を奪い合う状況に類似している。このことから、ヤマタノオロチが食事をする状況を用いてプロセススケジューリングの考え方を教えることが可能と判断した。なお、ヤマタノオロチは 8 つの頭を持つ 1 体の大蛇であるが、本アクティビティでは、便宜上、それぞれの頭をオロチと呼んでいる。コンピュータにおける技術要素とそれぞれのアンプラグド化について表 1 に整理する。

表 1 技術要素のアンプラグド化

技術要素	アンプラグド化
プロセス	オロチ
CPU 時間	食糧
プロセススケジューリング	オロチスケジューリング
プロセスの処理実行	オロチの食事
プロセスの処理時間	オロチの食事量
OS	学習者

(2) オロチのスケジューリングの進行状況を記入していくタイムチャートを用意する。その大きさは、机上で作業できるようにするため、A3 サイズである。スケジューリングは以下の手順で実行される。

- ① 何らかの方法で次に食事をするオロチを選ぶ。何らかの方法でそのオロチがその時に食べる食糧の量を決定する。一般的に、これらの「何らかの方法」はスケジューリングの方針に依存して決まる。
- ② 各オロチの食事の進捗状況をタイムチャートに記入していく。
- ③ これら 2 つ作業を、すべてのオロチが所定の量を食べ終わるまで繰り返す。
- ④ すべてのオロチが食事を終えた後、平均食事時間 (待ち時間を含む) と各オロチの応答時間 (そのオロチが食事をする状態になってから、実際に食事がありつくまでの待ち時間) を計算する。

図 1 に記入済み (実施済み) のタイムチャート例を示す。

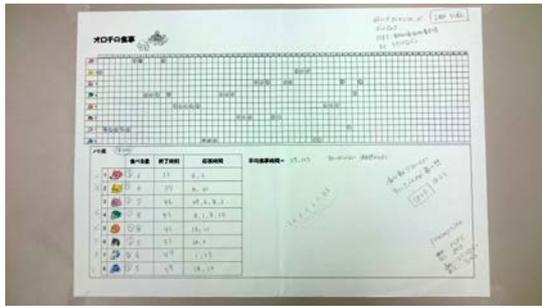


図1 記入済みのタイムチャート

(3) まず、練習ラウンドとして、ランダムなスケジューリング（以下、ランダムと略す）を行い、タイムチャートの記入方法などを学ぶ。例えば、8面ダイスを2回振って、次に食事を始めるオロチと、そのオロチがその時食べる量を決める。食べる量に応じて、タイムチャートにマークを記入する。これをすべてのオロチが食事を終えるまで繰り返す。

ポイントはある時刻に食事を始めるオロチとその時の食事量を、学習者の意向とは無関係に決めることにあり、8面ダイスを用いることが必須ではない。

応答時間は「あるオロチが食事を始める状態になってから、実際に食事にありつくまでの待ち時間」のことであり、1体のオロチについて複数個存在する場合がある。

(4) 次に、代表的なスケジューリングとしてFCFSをトップダウンに導入する。これは先着順（または、あらかじめ決められた順）に資源を割り当てるという最も単純なスケジューリングである。また、あるオロチが食事を始めると、そのオロチが食べるべき量をすべて食べ終わるまで、他のオロチがスケジューリングされないことを注意しておくことが重要である。これは、FCFSが横取りのないスケジューリングであることによる。

FCFSが先着順であることを説明した上でスケジューリングを行わせ、その結果をタイムチャートに記録させる。

(5) ランダムおよびFCFSの結果得られたタイムチャートにおける平均食事時間と応答時間に注意を向けさせ、スケジューリングに依存して、これらが異なることを確認させる。

これを受けて以下の2つの問題を提起し、その解を学習者に考えさせる。

① 問題1: 平均食事時間を最小にするようなスケジューリングを考案する。

② 問題2: 応答時間のばらつきを極力小さくするようなスケジューリングを考案する。

問題1のゴールは、SPTFである。これは食事量の少ないオロチから順にスケジューリングするものであり、横取りのないスケジューリングである。

いずれの問題においても、学習者たちだけで解答を見出すのが困難な場合には、適宜ヒントを与え、それぞれのゴールに導く。

最終的に、学習者には実際にSPTFとラウンドロビンに従ったスケジューリングを実施させる。その結果得られる平均食事時間と応答時間が、両スケジューリングの間でトレードオフの関係にあることを説明する。

(6) SPTFとラウンドロビンを体験したことで学習者の作業は終了し、後は座学形式でフォローアップを行う。フォローアップでは以下の点を強調し、スケジューリングに関する知識の定着を図る。

① 方針に応じて多様なスケジューリングが考えられること。

② 平均処理時間（平均食事時間）の短縮と応答時間のばらつきの抑制は両立しないこと。

③ スケジューリングには横取りのあるタイプと横取りのないタイプがあること。

④ 実際のOS（Windows, Android等々）がプロセススケジューリングを行う機能を有すること。

(7) 本アクティビティを高等専門学校の低学年（2年生、3年生）や大学の新入生に対して試行する。試行後のアンケートの結果から、本アクティビティの実用性ならびに有効性を確認する。

4. 研究成果

(1) 西田等が提案するCSアンブレグドのデザインパターンには、学習法としての良さにつながる特徴的なパターンが整理されている。同デザインパターンにおける、設計段階で考慮すべき項目に関する対応状況について整理する。

① 問題状況 (Problem)

・コンピュータ科学を理解することが難しい。
→ OSの学習においてプロセススケジューリングは必須の学習項目である。

② 問題背景 (Context)

・専門知識を持たない子ども（7から15歳程度）でも理解できる必要がある。

→ OSというものの存在については知っている学習者を前提とする。また、スキルとしては平均値の計算ができることのみを前提とするが、コンピュータ分野特有の略語（アルファベット数文字の場合が多い）を導入することから中学生以上を対象とする。

③ フォース (Forces)

・コンピュータを使わずに教えられる。

→ 学習者の作業はタイムチャートへの記入が主であり、コンピュータを使うことはない。

・専門家でない教員が教えられる。

→ 学習者の作業進行を指導するだけであれば専門知識を必要としない。しかし、実際のコンピュータでの利用の解説までを考えると、OSに関する知識を有する教員であることが望ましい。

・20～40分で学ぶことができる。

→ 90分授業1回または50分授業2回を想定して設計している。

④ 解決策 (Solution)

・学習にゲームの要素を取り入れる。

→ スケジューリング作業そのものにゲーム性はない。SPTFやラウンドロビンを自ら考案する段階でチーム間での競争を取り入れ、チャレンジ性を確保している。

・教える対象ごとに適した教材を使う。

→ 今回想定した試行対象（高専低学年生、大学新入生）に関しては、適していると判断する。

・体験による学習を行う。

→ スケジューリングの進行状況をタイムチャートに記入することによって、体験的な学習となっている。

・体験の中で試行錯誤しながら考えさせる。

→ SPTFやラウンドロビンを考案させることで、「試行錯誤しながら考える」という状況を作り出している。

・グループワークによる共同作業を取り入れる。

→ グループでの共同作業を前提とする。

・ワークシートを用意する。

→ スケジューリングの進行状況を記入するタイムチャートを用意する。

・教具を手軽で安価に作成できる。

→ 用意するのは基本的にタイムチャートのみであり、必要に応じて8面ダイス等も用意する。いずれも、安価に用意できるものである。

・さまざまな場所で行える。

→ タイムチャートに記入できる環境があればどこでも実施できる。

・実際のコンピュータでの利用を解説する。

→ OSにおけるプロセススケジューリングとの対応を解説したスライドを用意する。

⑤ 根拠 (Rationale)

・体験する内容はコンピュータ科学のエッセンスから構成されている。

→ プロセススケジューリングはコンピュータ科学のエッセンスである。

(2) 本アクティビティの試行を3回行った。各試行の諸元を表2に示す。

表2 各試行の諸元

第1回試行	
日時	2014年8月4日(月)
場所	松江高専(松江市)
対象者	情報工学科3年生
人数	16人(男12, 女4)
体制	4チーム
年齢	17~18歳
第2回試行	
日時	2015年2月17日(火)
場所	松江高専(松江市)
対象者	情報工学科2年生
人数	16人(男10, 女6)
体制	4チーム

年齢	16~17歳
第3回試行	
日時	2015年4月9日(木)
場所	東京電機大学 (千葉県印西市)
対象者	情報環境学部1年生
人数	5人(男5)
体制	5チーム
年齢	18~19歳

試行時の様子を図2に示す。

各試行において、作業終了後アンケートによるフィードバックを得た。アンケート内容はすべての施行において同じである。



図2 試行の様子

アンケートは13個の選択型設問と1個の自由記述型設問からなる。選択型設問のうち12個は4択設問、1個は3択設問である。

主なアンケート項目を以下に示す。

- ① スケジューリングの認知度
今回の説明を受ける以前から、スケジューリングについて知っていたか。
- ② スケジューリングに関する事後理解
スケジューリングに関する事前知識がない場合、今回の作業を通して理解できたか。
- ③ スケジューリングの表現性
本アクティビティはスケジューリングをうまく表現できているか。
- ④ 具体的スケジューリングの理解
各スケジューリング(FCFS, SPTF, ラウンドロビン)とその特徴を理解できたか。
- ⑤ 親しみやすさ
ヤマタノオロチを使うことに親しみを感じたか。

これらの結果を表3から表7に示す。表6以外は各試行のアンケートにおいて得られ回答数をそのまま掲載している。表6では、「よく分かった」に4点、「まあまあ分かった」に3点、「あまり分からなかった」に2点および「まったく分からなかった」に1点を与え、それをそれぞれの選択肢に対する回答数によって荷重平均した得点を求めた。

表3および表4からは、ほとんどの学習者がスケジューリングに関する事前知識を持っていなかったにもかかわらず、試行の結果、ほぼ全員がそれについて理解できたことが分かる。

また表5からは、本アクティビティがプロ

表3 認知度

選択肢	第1回	第2回	第3回
よく知っていた	0	0	0
少しは知っていた	2	0	0
詳しくは知らなかった	5	3	1
まったく知らなかった	9	13	4

表4 事後理解

選択肢	第1回	第2回	第3回
理解できた	12	9	3
まあまあ理解できた	2	7	1
あまり理解できなかった	0	0	1
理解できなかった	0	0	0

表5 表現性

選択肢	第1回	第2回	第3回
表現できている	13	8	2
まあまあ表現できている	3	8	2
あまり表現できていない	0	0	1
表現できていない	0	0	0

表6 各技法の理解 (4点満点)

技法名称	第1回	第2回	第3回
FCFS	3.81	3.81	3.60
SPTF	3.88	3.88	3.60
ラウンドロビン	3.81	3.75	3.40

表7 親しみやすさ

選択肢	第1回	第2回	第3回
大いに感じた	4	4	0
まあまあ感じた	6	7	1
あまり感じなかった	4	4	3
まったく感じなかった	2	1	1

セススケジューリングをうまく表現できているかという点についても肯定的な評価を得たと判断する。

表6からは、今回扱った3種類の具体的スケジューリングに関して、その理解度に大差はなく、いずれも高い理解度を得たと判断する。

表7からは、ヤマタノオロチという地域素材を用いたことが、「親しみやすさ」という観点から一定の効果があったことも確認できた。

(3) デザインパターンに従って、試行内容と試行結果に関する検証を行う。

① 解決策 (Solution)

- ・教える対象ごとに適した教具を使う。

→ アンケート項目「スケジューリングの表現性」において高い評価が得られたことから、今回の試行対象に関しては、適した教具であったと判断する。

② 問題解決後の状況 (Resulting Context)

- ・体験した特定の状況は理解している。

→ アンケート項目「スケジューリングの認知度」と「スケジューリングに関する事後理解」の結果から判断して、オロチの食事におけるスケジューリングという特定の状況については理解できたと考える。

- ・体験した内容とコンピュータ科学の原理とは漠然とした対応しかない。

→ フォローアップにおいてコンピュータにおけるプロセススケジューリングとの対応について解説し、理解を得たと考える。

- ・コンピュータ科学のその領域のうち、一部分だけが扱われている。

→ 本アクティビティで扱えていないプロセススケジューリングもある。

③ 根拠 (Rationale)

- ・共同作業とゲームの要素により意欲が向上し、コンピュータ科学の内容を真剣に考える。

→ ゲーム性がやや弱かったにもかかわらず、意欲的に作業に取り組んでいた。スケジューリングの重要性についてはしっかりと理解してくれたと判断する。

- ・集中して考える体験により、体験した内容が定着する。

→ アンケート項目「スケジューリングに関する事後理解」において高い評価を得ることができたことから、体験した内容の定着が期待できる。

(4) 以上のデザインパターンによる検証、ならびにアンケート結果から、本アクティビティはプロセススケジューリングの体験的導入教育の教材として成立するものと判断する。

5. 主な発表論文等

〔雑誌論文〕(計1件)

① Hisao Fukuoka, Akane Kawakami, Yuji Wada, A Proposal for CS Unplugged Utilizing Regional Materials, International Journal of Informatics Society, 査読有, Vol.7, No.1, pp.13-18, June 2015.
http://www.infsoc.org/journal/vol07/IJIS_07_1_013-018.pdf

〔学会発表〕(計9件)

① 福岡久雄, 渡部徹, 廣瀬誠, 小川仁士, CS アンプラグド“オロチの食事”の提案と試行—プロセススケジューリングを学ぶ—, 情報処理学会 情報教育シンポジウム Summer Symposium in Sakaiminato 2015 (SSS2015) 論文集, 査読有, pp.215-222, 2015年8月, 境港マリーナホテル(鳥取県境港市).

② 福岡久雄, 渡部徹, 廣瀬誠, 小川仁士,
オロチの食事ースケジューリング技法を学
習するためのCSアンプラグドActivity, 情
報処理学会研究報告(コンピュータと教育研
究会), 査読無, Vol.2014-CE-127, No.6,
pp.1-7, 2014年12月, 九州大学大橋キャン
パス(福岡県福岡市).

③ Hisao Fukuoka, Akane Kawakami, Yuji
Wada, A Proposal for CS Unplugged
Utilizing Regional Materials,
International Workshop on Informatics
(IWIN2014), 査読有, pp.109-114, September
2014, プラハ(チェコ共和国).

6. 研究組織

(1) 研究代表者

福岡 久雄 (FUKUOKA, Hisao)

東京電機大学・情報環境学部・特別専任教
授

研究者番号: 10370016