

मॉड्यूल 4 - अभिकलनात्मक सोच या कम्प्यूटेशनल थिंकिंग

सीखने का उद्देश्य	सीख के परिणाम
<ul style="list-style-type: none">● छात्रों को पर्याप्त प्रौद्योगिकी का उपयोग करके अपने विचारों को एक मजबूत चरण-दर-चरण समाधान के रूप में विकसित करने में मदद करना● उन्हें अभिकलनात्मक सोच की अवधारणा से परिचित कराना - इसकी प्रमुख तकनीक / दृष्टिकोण● व्यावहारिक व क्रियाशील या डीआईवाई गतिविधियों एवं एक्सपोजर के माध्यम से उन्हें एल्गोरिदम या कलन विधि, फ्लोचार्ट, सेंसर, डेवलपमेंट बोर्ड, प्रोग्रामिंग आदि के बारे में समझाना	<ul style="list-style-type: none">● समस्या समाधान के लिए अभिकलनात्मक सोच के सिद्धांत और इसके प्रयोग के बारे में समझ जाएंगे● अपने विचारों को फ्लोचार्ट, एल्गोरिदम में विकसित करना और उन्हें प्रोग्रामिंग भाषा में परिवर्तित करना जान जाएंगे सेंसर, कोडिंग का उपयोग करते हुए वास्तविक व्यावहारिक● परियोजनाओं से गुजरते हुए उनके अनुप्रयोगों को समझ जाएंगे

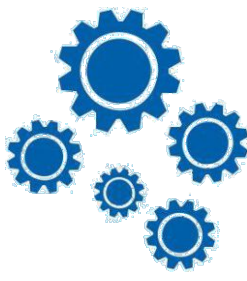
अभिकलनात्मक सोच के बारे में

कंप्यूटर का उपयोग समस्याओं को हल करने में हमारी सहायता के लिए किया जा सकता है। हालाँकि, किसी समस्या से निपटने से पहले, समस्या और उसे हल करने के तरीकों को समझने की आवश्यकता है। अभिकलनात्मक सोच हमें ऐसा करने की अनुमति देती है। यह हमें एक जटिल समस्या लेने, उसे समझने और संभव समाधान विकसित करने की अनुमति देता है। ये समाधान तब इस तरह से प्रस्तुत किए जा सकते हैं जैसे एक कंप्यूटर और एक मानव दोनों इसे समझ सकते हैं।

अभिकलनात्मक रूप से सोचना प्रोग्रामिंग नहीं है। यह कंप्यूटर की तरह सोच भी नहीं रहा है! सीधे शब्दों में कहें, प्रोग्रामिंग एक कंप्यूटर को बताता है कि क्या करना है और कैसे करना है। हमें कंप्यूटर को क्या करने के लिए बोलना है अभिकलनात्मक सोच हमें उसे तैयार करने में मदद करती है।

उदाहरण के लिए, यदि आप अपने दोस्तों से ऐसी जगह पर मिलने के लिए तैयार होते हैं जहाँ आप पहले कभी नहीं गए हैं, तो आप अपने घर से बाहर कदम रखने से पहले आपको अपने मार्ग की योजना बनानी होगी। आप उपलब्ध मार्गों पर विचार कर सकते हैं और कौन सा मार्ग - 'सबसे अच्छा है' - यह वह मार्ग हो सकता है जो सबसे छोटा, सबसे तेज हो, या वह हो जो रास्ते में आपकी पसंदीदा दुकान से गुजरे। फिर आप वहाँ पहुंचने के लिए कदम-दर-कदम निर्देशों का पालन करेंगे। इस मामले में, योजना बनाने का चरण अभिकलनात्मक सोच की तरह है, और निर्देशों का पालन करना प्रोग्रामिंग की तरह है।

एक जटिल समस्या को एक आसानी से समझ में आने वाली समस्या के रूप में बदलने का कौशल अत्यंत उपयोगी है। वास्तव में, यह एक ऐसा कौशल है जो आपके पास पहले से है और शायद आप इसे हर दिन उपयोग करते हैं।



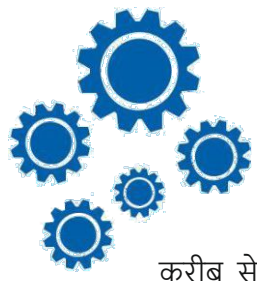
आइए दो और उदाहरण देखें :

उदाहरण 1	उदाहरण 2
<p>मान लें कि आपको यह तय करने की आवश्यकता है कि आपके दोस्तों के समूह के साथ क्या करना है। यदि आप सभी को अलग-अलग चीजें पसंद हैं, तो आपको यह तय करना होगा:</p> <ul style="list-style-type: none">• आप क्या कर सकते थे• आप कहाँ जा सकते थे• कौन क्या करना चाहता है• आपने पहले क्या किया है जो पिछली बार सफल रहा है• आपके पास कितना पैसा है और किसी भी विकल्प के लिए कितनी लागत है• आपके पास कितना समय है <p>इस जानकारी से, आप और आपके मित्र अधिक आसानी से तय कर सकते हैं कि कहाँ जाना है और क्या करना है - ताकि आपके अधिकांश दोस्त खुश रहें। आप आज भी और यहां तक कि भविष्य में भी समस्या का सबसे अच्छा समाधान तैयार करने के लिए आंकड़े एकत्र करने और उसका विश्लेषण करने में आपकी सहायता करने के लिए एक कंप्यूटर का उपयोग कर सकते हैं।</p>	<p>एक अन्य उदाहरण हो सकता है जब एक वीडियो गेम खेला जा रहा हो। खेल के मुताबिक एक स्तर को पूरा करने के लिए आपको जानना होगा:</p> <ul style="list-style-type: none">• आपको किन वस्तुओं को इकट्ठा करने की आवश्यकता है, आप उन्हें कैसे एकत्र कर सकते हैं और उन्हें एकत्रित करने के लिए आपके पास कितना समय है• त्वरित समय में स्तर को पूरा करने के लिए सबसे अच्छा मार्ग क्या है• किस तरह के दुश्मन हैं और उनकी कमजोरियाँ क्या हैं <p>इन विवरणों से आप सबसे कुशल तरीके से स्तर को पूरा करने की रणनीति बना सकते हैं।</p>

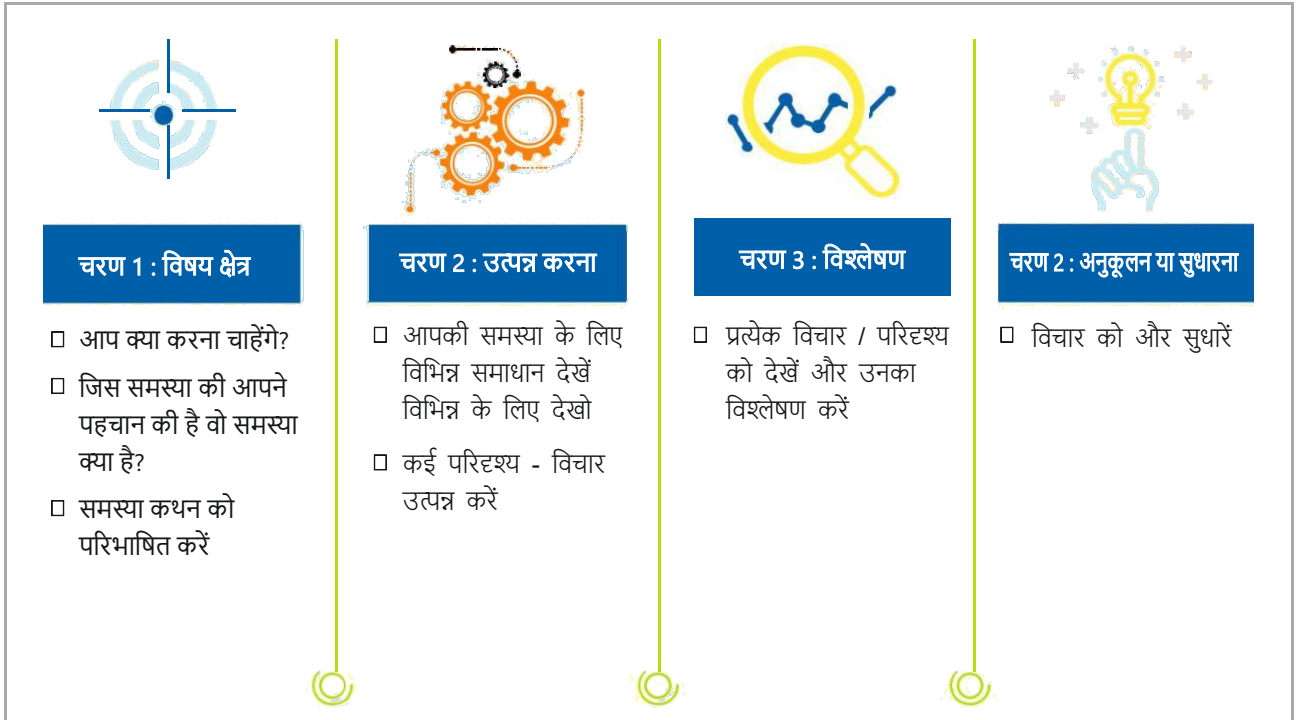
उपरोक्त दोनों उदाहरणों से पता चलता है कि जटिल समस्या को हल करने के लिए अभिकलनात्मक सोच का उपयोग कैसे किया जाता है।

यह हमें अभिकलनात्मक सोच की चार प्रमुख तकनीकों कि ओर लाता है:

1. **अपघटन** - एक जटिल समस्या या प्रणाली को छोटे, अधिक प्रबंधनीय भागों में तोड़ना (जैसे कि कहाँ जाना है, स्तर कैसे पूरा करें)
2. **अमूर्तता** - केवल महत्वपूर्ण जानकारी पर ध्यान केंद्रित करना, अप्रासंगिक विवरण (जैसे मौसम, बाहर निकलने का स्थान) की अनदेखी करना
3. **पैटर्न को पहचानना** - समस्याओं के बीच और उनके अंदर समानताओं की तलाश करना (जैसे पिछली समान समस्याओं के उपयोग का ज्ञान)
4. **एल्गोरिदम** - समस्या का एक कदम-दर-कदम समाधान विकसित करना, या समस्या को हल करने के लिए नियमों का पालन करना (उदाहरण के लिए कदम-दर-कदम कार्य योजना तैयार करना)



करीब से देखने पर पता चलेगा कि आपने इन तकनीकों के कुछ हिस्सों का उपयोग विचार करने और डिज़ाइन के बारे में सोचने के मॉड्यूल में काम किया होगा। यहाँ एक त्वरित सैपशॉट है:



इसलिए, इस मॉड्यूल में हम इनपर ध्यान केंद्रित करेंगे

1. एल्गोरिदम और फ़्लोचार्ट - समस्या के समाधान के लिए चरण-दर-चरण समाधान विकसित करना, या समस्या को हल करने के लिए नियमों का पालन करना।
2. हार्डवेयर / सॉफ्टवेयर से परिचित होना - उनके उपयोग को समझना और विचार करने के माध्यम से उत्पन्न विचार को रखना और डिज़ाइन के माध्यम से विस्तृत करना सीखें, इसे कंप्यूटर पर ले जाकर इस पर काम करना और इसे वास्तविक आकार देना सीखना
3. प्रोग्रामिंग के मूल सिद्धांत को सीखना

एल्गोरिदम और फ़्लोचार्ट को समझना

एल्गोरिदम एक योजना है, एक समस्या को हल करने के लिए चरण-दर-चरण निर्देशों का एक संग्रह है।

यदि आप जूते के फीते को बांध सकते हैं, एक कप चाय बना सकते हैं, कपड़े पहन सकते हैं या भोजन तैयार कर सकते हैं, तो आप पहले से ही जानते हैं कि एक एल्गोरिदम का पालन कैसे करें। एक एल्गोरिदम में प्रत्येक निर्देश की पहचान की जाती है और जिस क्रम में उन्हें पूरा किया जाना चाहिए उसकी योजना बनाई जाती है। कंप्यूटर प्रोग्राम बनाने के लिए शुरुआती कार्य के रूप में अक्सर एल्गोरिदम का उपयोग किया जाता है, और उन्हें कभी-कभी फ़्लोचार्ट के रूप में लिखा जाता है।

यदि हम कंप्यूटर को कुछ करने के लिए कहना चाहते हैं, तो हमें एक कंप्यूटर प्रोग्राम लिखना होगा जो कंप्यूटर को चरण-दर-चरण निर्देश देगा कि हम इससे कैसे और क्या करवाना चाहते हैं। इस चरण-दर-चरण कार्यक्रम को योजना की आवश्यकता होगी, और ऐसा करने के लिए हम एक एल्गोरिदम का उपयोग करते हैं।

कंप्यूटर केवल उतने ही अच्छे होते हैं जितना कि उनके द्वारा दिए गए एल्गोरिदम। यदि आप कंप्यूटर को खराब एल्गोरिदम देते हैं, तो आपको खराब परिणाम मिलेगा। गणना, आंकड़ों का प्रसंस्करण और स्वचालन सहित कई अलग-अलग चीजों के लिए एल्गोरिदम का उपयोग किया जाता है।



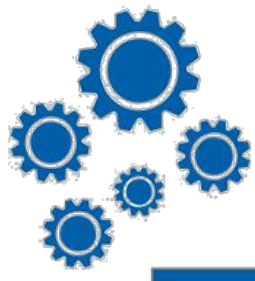
एक अल्गोरिदम का वर्णन: फ़्लोचार्ट्स

फ़्लोचार्ट एक आरेख है जो निर्देशों के एक संग्रह को दिखाता है। फ़्लोचार्ट आमतौर पर विभिन्न निर्देशों को दिखाने के लिए मानक प्रतीकों का उपयोग करते हैं। फ़्लोचार्ट में किस स्तर का विस्तृत विवरण आवश्यक है इसके बारे में कुछ नियम हैं। कभी-कभी फ़्लोचार्ट को कई चरणों में विभाजित किया जाता है ताकि जो वास्तव में हो रहा है इसके बारे में बहुत विस्तृत प्रदान किया जा सके। कभी-कभी उन्हें सरलीकृत किया जाता है, ताकि केवल एक ही कदम में कई चरण शामिल होते हैं।

किसी से उनका नाम और उम्र पूछने के लिए एवं नीचे दर्शाये गए प्रोग्राम के आधार पर एक टिपण्णी करने ले लिए जो फ़्लोचार्ट होगा वो कुछ ऐसा नज़र आएगा:



Source: <http://www.bbc.co.uk/education/guides/zpp49j6/revision/3>



सहजकर्ता के लिए नोट

- कृपया सुनिश्चित करें की सभी प्रतिभागी / छात्र प्रस्तुति के अंत में दिए गए अभ्यासों को हल करें। छात्रों को फ्लोचार्ट और एल्गोरिदम के माध्यम से समस्याओं और समाधान के साथ आने के लिए प्रोत्साहित किया जा सकता है।

सेंसर और एक्चुएटर्स या प्रवर्तकों, और उनके अनुप्रयोगों को समझना

यहां आप विभिन्न प्रकार के सेंसर और उनके अनुप्रयोगों के बारे में जानेंगे।

मानव शरीर को उसके पांच मूल इंद्रियों के बिना कल्पना करें। यह अपने परिवेश के साथ बातचीत नहीं कर पाएगा, और यह किसी भी प्रतिक्रिया देने में सक्षम नहीं होगा। इसी प्रकार कंप्यूटर की दुनिया में हमारे पास सेंसर हैं - किसी भी चयनित प्रक्रिया / कार्यक्रम के लिए इनपुट / आउटपुट को जानने में मदद करने के लिए।

व्यापक परिभाषा में, एक सेंसर एक इलेक्ट्रॉनिक घटक, मॉड्यूल, या उपतंत्र है जिसका उद्देश्य अपने वातावरण में घटनाओं या परिवर्तनों का पता लगाना है, और इस जानकारी को अन्य विद्युत् उपकरणों, अक्सर एक कंप्यूटर प्रोसेसर को भेजना है। एक सेंसर हमेशा अन्य विद्युत् उपकरणों के साथ प्रयोग किया जाता है, चाहे एक बिजली से चलने वाले लाइट के रूप में या कंप्यूटर जैसी जटिल चीज़ के रूप में हो।

सेंसर विभिन्न प्रकार के आकार और लम्बाई-चौड़ाई में आते हैं, और वे कई प्रकार की चीज़ों को महसूस कर सकते हैं, दिल की धड़कन से लेकर हवा के दबाव तक, चमक से लेकर गर्मी तक - लगभग हर चीज़ को मापने के लिए एक सेंसर होता है। और यदि आपकी आवश्यकता के लिए कोई सेंसर विशिष्ट नहीं है, तो आप अपनी आवश्यकता को पूरा करने के लिए कई प्रकार के सेंसर को मिला सकते हैं या उनका मेल कर सकते हैं।

प्रवर्तक मूल रूप से कोई कार्रवाई उत्पन्न करने वाली चीज़ है। वे किसी व्यक्ति के हाथों और पैरों की तरह हैं, और वे विभिन्न रूपों और आकारों में आते हैं। इलेक्ट्रोमैग्नेट्स, रिले, डीसी मोटर्स, सर्वो मोटर्स आदि आमतौर पर इस्तेमाल किए जाने वाले प्रवर्तकों के कुछ उदाहरण हैं।

सहजकर्ता के लिए नोट

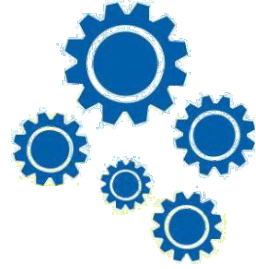
- कृपया पेन ड्राइव में दिए गए सेंसर और प्रवर्तकों के बारे में अभिकलनात्मक सोच प्रेजेंटेशन देखें
- कृपया सुनिश्चित करें कि सभी प्रतिभागियों / छात्रों ने प्रस्तुति के अंत में दिए गए अभ्यासों को हल किया हो। छात्रों को फ्लोचार्ट और एल्गोरिदम के माध्यम से समस्याओं और समाधान के साथ आने के लिए प्रोत्साहित किया जा सकता है।

सर्किट को समझना

इससे आपको यह समझने में मदद मिलेगी कि सर्किट क्या है और विभिन्न प्रकार के सर्किट कैसे होते हैं।

इलेक्ट्रॉनिक्स में एक सर्किट दो या दो से अधिक बिंदुओं के बीच का एक मार्ग है जिसके बीच विद्युत् प्रवाह किया जा सकता है। एक सर्किट एक बंद लूप या फंदा होता है जिसमें से इलेक्ट्रॉन यात्रा कर सकते हैं। बिजली का एक स्रोत, जैसे कि बैटरी, सर्किट में विद्युत् ऊर्जा प्रदान करता है। जब तक सर्किट पूरा नहीं हो जाता है, अर्थात्, विद्युत् स्रोत तक वापस जाकर एक पूर्ण चक्र नहीं बना देता, तब तक कोई भी इलेक्ट्रॉन अपनी जगह से नहीं हिलेगा।³

<http://www.qrg.northwestern.edu/projects/vss/docs/power/2/whats a circuit.ht>



संसाधन

कुछ और प्रेरणा पाने के लिए आप निम्नलिखित को भी देख सकते हैं:

संक्षिप्त विवरण	रिसोर्स यूआरएल
कम्प्यूटेशनल थिंकिंग और थिंकिंग कम्प्यूटिंग के बारे में	https://www.cs.cmu.edu/afs/cs/usr/wing/www/talks/stand-tc-long.pdf
कम्प्यूटेशनल थिंकिंग के बारे में ज्यादा जानकारी	Computational Thinking (In depth Explanation) https://www.youtube.com/watch?v=C2Pq4NiE4I Computational Thinking from code.org https://www.youtube.com/watch?v=injJWiSA0pw Abstraction Computational Thinking https://www.youtube.com/watch?v=jV-7Hy-PF2Q
एकल बोर्ड माइक्रोकंट्रोलर के बारे में ज्यादा जानकारी	https://en.wikipedia.org/wiki/Singleboard_microcontroller
सर्किट के बारे में	http://www.build-electronic-circuits.com/free-electronic-circuits/ http://www.dummies.com/programming/electronics/how-to-build-a-simple-electronic-circuit/ Basic Electricity :Current ,Resistance and Ohms law https://www.youtube.com/watch?v=NfcgA1axPlo Basic Tutorial on Introduction to Circuits by Khan Academy (Tutorial) https://www.youtube.com/watch?v=3o8_EARoMtg There is a course on Circuit Physics by Khan Academy. It's Very Good for Beginners. (Playlist for Circuit Physics Course) https://www.khanacademy.org/science/physics/circuits-topic How to use Breadboard https://www.youtube.com/watch?v=6WReFkfrUlk Paper Circuit (How to do it) https://www.youtube.com/watch?v=BwKQ9Idq9FM Build your own USB charger - a step by step guide http://www.build-electronic-circuits.com/wp-content/uploads/2017/1m/How-To-Build-A-Portable-USB-Charger.pdf
सेंसर्स के बारे में	https://www.electrical4u.com/sensor-types-of-sensor/ https://www.engineersgarage.com/articles/sensors https://www.youtube.com/watch?v=q1xNuU7gaAQ
अल्गोरिदम क्या है?	https://www.youtube.com/watch?v=6hfOvs8pY1k https://www.youtube.com/watch?v=Da5TOXCwLsg
अल्गोरिदम इन सुडो कोड, फ्लो डायग्राम्स एंड प्रोग्रामिंग एन ओवरव्यू ऑफ़ फ्लो चार्ट	https://www.youtube.com/watch?v=HhBrkpTqzqg https://www.youtube.com/watch?v=XDWw4Ltfy5w
एन ओवरव्यू ऑफ़ फ्लो चार्ट	https://www.youtube.com/watch?v=uCNliFuKG8I
रियल लाइफ अल्गोरिदम-पेपर एयरप्लेनेस	https://www.youtube.com/watch?v=AWqo8Gxtrjs
इलेक्ट्रॉनिक कंपोनेंट्स	Simple Guide to Electronics (Tutorial) https://www.youtube.com/watch?v=6Maq5lyHSuc Basics of Coils (Part 1) (Tutorial) https://www.youtube.com/watch?v=kdrP9WbJlb8 Basics of Coils (Part 2) (Tutorial) https://www.youtube.com/watch?v=XCnI6ZOYKes How Does a Transistor Work? (Tutorial) https://www.youtube.com/watch?v=IcrBqCFLHIY
मेकिंग द मोटर्स वर्क डी.सी. मोटर वर्किंग (ट्यूटोरियल) स्टेपर मोटर वर्किंग (ट्यूटोरियल)	D.C Motor Working (Tutorial) https://www.youtube.com/watch?v=LAtPHANefQo&t=65s Stepper Motor Working (Tutorial) https://www.youtube.com/watch?v=TWMai3oirnM