

STAGE DI INFORMATICA 2009/2010 al Centro SIAF (Volterra)

CALENDARIO ATTIVITA' dal 01/02/2010 al 06/02/2010

Program. giornaliera	Data	dalle ore	alle ore	Totale ore	Finalità/Obiettivi	Contenuti	Metodologia (Teoria, Esercitazione Pratica, Laboratorio)
	01/02/10	08:00	09:00	1	Presentare subito il sito e l'archivio dei problemi risulta utile non soltanto perché l'archivio dei problemi è lo strumento principale per prepararsi alle iOi anche in autonomia, fuori dalla Learning Week ed oltre ad essa, ma coglie anche la finalità di porre subito delle sfide di problem solving, e di dire subito che questo è il taglio di questa settimana.	Presentazione del sito AICA per le iOi ed in particolare dell'archivio problemi. Scelta di un problema.	Laboratorio
	01/02/10	09:00	10:00	1	Lo sfidare i ragazzi con dei problemi coglie la finalità di cui sopra. Inoltre ci fornisce assai presto una griglia di ingresso, ossia ci da un quadro delle conoscenze e competenze pregresse ben mirato alle iOi come anche alla Learning Week.	Piccoli problemi di programmazione per una verifica della situazione iniziale dei ragazzi. Probabilmente questi problemi saranno scelti tra quelli del sito AICA eventualmente dai ragazzi stessi guidati su una rosa di problemi adatti per partire. Un problema adatto per partire deve comunque stimolare il ragazzo coinvolgendo le sue capacità di problem solving, ma deve risultare facile abbastanza da non lasciare indietro nessuno e da consentirci di verificare le competenze in materia di programmazione senza consentire che eventuali carenze su questo piano possano risultare bloccanti. L'implementazione deve quindi richiedere poche righe di codice.	Esercitazione pratica
	01/02/10	10:00	11:00	1	I ragazzi devono ambientarsi e sperimentare linguaggio, macchina virtuale, ed ambiente di programmazione.	Scrittura individuale di semplici programmi che svolgano dei compiti proposti centralmente.	Esercitazione pratica
					I ragazzi devono raggiungere una conoscenza adeguata delle possibilità offerte da un linguaggio ad alto livello e di come ne vada fatto buon uso. Ovviamente è importante riallineare i ragazzi e coprire le lacune fornendo le competenze necessarie al muoversi.	Affinamento di alcuni programmi già scritti. Presentazione di alcune caratteristiche più avanzate del linguaggio ad alto livello. Scrittura congiunta di semplici programmi. Assestamento di eventuali nozioni/competenze in materia di programmazione. Prendendo spunto dai codici prodotti dai ragazzi, se ne esegue una rilettura critica, confrontando i punti di forza delle varie implementazioni. Si cerca di approfittare di questa occasione per dare dei segnali chiari sull'attenzione alla pulizia e leggibilità del codice. E si cerca di capire se sia il caso di introdurre o rivedere nozioni e/o tecniche in materia di programmazione.	Teoria e Esercitazione pratica
1° giorno	25/01/10	11:00	12:00	1			

pausa pranzo						
01/02/10	14:00	15:00	1	Se si prosegue con gli approfondimenti in materia di programmazione le finalita' sono sostanzialmente quelle di cui sopra, e di assestare una piattaforma comune di intesa e standardizzazione. Se ci si rivolge sul versante algoritmico si va a cogliere al cuore quelle che sono le competenze che il corso mira a sviluppare e promuovere.	Difficile prevedere ora se quest'ora sara' meglio dedicata ad approfondimenti sul piano della programmazione oppure se convenga portarsi sul piano degli algoritmi. Nel primo caso si potrebbe lavorare su alcune strutture dati di base. Nel secondo caso si incomincia invece a spendere considerazioni di efficienza ed un argomento che potrebbe risultare strategico potrebbe essere quello degli Algoritmi di ordinamento. Una soluzione ibrida potrebbe essere quella di proporre ai ragazzi il problema del sottointervallo di somma massima, problema che potrebbe poi essere richiamato in vari momenti ed utilizzato come banco di prova di varie tecniche algoritmiche via e via piu' potenti.	Esercitazione pratica
01/02/10	15:00	16:00	1	far capire ai ragazzi cosa e' un algoritmo. Far capire ai ragazzi che loro sanno inventarsi algoritmi, che li hanno sempre fatto. Rendere consapevoli i ragazzi della pervasivita' degli algoritmi e della loro rilevanza.	Algoritmi di Ordinamento. Si chiede di sviluppare ai ragazzi strategie per ordinare dei pesi confrontabili (modello bilancia a braccia eguali). Si cerca di dare un nome ai vari approcci proposti dai ragazzi e di collocarli nella vasta letteratura o per metodologia generale. Si considera l'efficienza delle soluzioni proposte.	Teoria e Esercitazione pratica
01/02/10	16:00	17:00	1	Implementare porta a consolidare. Di fatto solo con l'implementazione siamo sicuri della piena comprensione. Saper mettere in pratica e saper codificare sotto pressione sono abilita' necessarie per le gare. Lo studente riceve feedback su svariati livelli e da piu' soggetti. Con cio' cogliamo inoltre la finalita' metacognitive e di autovalutazione. Il lavoro autonomo risulta fondamentale per le Olimpiadi. Anche al di la del guardare alle Olimpiadi il corso si pone la finalita' di stimolare i ragazzi e di condurli ad una maggiore motivazione, consapevolezza ed autonomia.	Implementazione da parte dei ragazzi degli algoritmi analizzati assieme. Proposta ai ragazzi di alcuni problemi per approfondimento autonomo (fuori dalle ore di lezione) dal sito delle Olimpiadi. Ascolto se i ragazzi hanno alcuni problemi di loro interesse da affrontare assieme a lezione o negli altri spazi della giornata. Proseguiamo per problemi e proponiamo giochi (solitario francese, forza 4, mastermind, cubo di Rubik, 15).	Esercitazione pratica
02/02/10	08:00	09:00	1	L'induzione matematica, oltre all'importanza fondamentale in matematica e come strumento di dimostrazione e pensiero, va introdotta per poter più congruamente introdurre la ricorsione.	Introduzione all'induzione matematica	Teoria
02/02/10	09:00	10:00	1	L'induzione matematica, oltre all'importanza fondamentale in matematica e come strumento di dimostrazione e pensiero, va introdotta per poter più congruamente introdurre la ricorsione.	L'induzione matematica	Teoria

2° giorno	02/02/10	10:00	11:00	1	Sviluppo della ricorsione quale tecnica di problem solving.	Introduzione alla ricorsione – frontale. Probabilmente si propone il rompicapo della torre di Hanoi e si cerca di raccogliere assieme, guidati dall'insegnante alla lavagna, un modo sound di procedere nell'esplorazione e soluzione del rompicapo e di questioni ad esso collegate. Il messaggio che e' prioritario far pervenire in modo chiaro e' la potenza del pensiero induttivo come strumento di problem solving. La ricorsione viene introdotta come una forma di astrazione e di salto ad un metalivello.	Teoria e Esercitazione pratica	
	02/02/10	11:00	12:00	1	Sviluppo del concetto di ricorsione, quale fondamento in informatica teorica nello sviluppo delle procedure ricorsive nella programmazione	Introduzione alla ricorsione – per problemi. I ragazzi devono ora codificare la soluzione ricorsiva raccolta per il problema precedentemente esplorato., probabilmente quello della torre di Hanoi. Per quelli di loro che non avessero conoscenza della possibilita' e specificita' delle chiamate ricorsive previste dal linguaggio di programmazione scelto, si fara' qualche esempio semplice, probabilmente basato sul computo di n fattoriale. Ma poi sara' un'esperienza importante la codifica ricorsiva non banale del problema della torre di Hanoi, dove gli studenti avranno modo di toccare con mano l'importanza di una fiducia piena al salto di metalivello che si e' chiamati ad abbracciare per scrivere procedure ricorsive in scioltezza.	Esercitazione pratica	
	02/02/10	pausa pranzo						
	02/02/10	14:00	15:00	1	e' bene che gli studenti abbiano un quadro della portata di ricorsione ed induzione nel pensiero scientifico-matematico moderno. E' fondamentale che gli studenti si mettano alla prova con la ricorsione,	Introduzione alla ricorsione – per problemi. Probabilmente saremo presi dal portare avanti l'implementazione detta sopra. In caso contrario si danno esempi dell'uso dell'induzione in matematica. Si considerano e propongono problemi di conteggio che portino alla formulazione di ricorrenze.	Esercitazione pratica	
	02/02/10	15:00	16:00	1	E' fondamentale che gli studenti si mettano alla prova con la ricorsione,	Introduzione alla ricorsione – per problemi. Si proseguono su problemi che portino alla scoperta di ricorrenze anche classiche.	Esercitazione pratica	
	02/02/10	16:00	17:00	1	e' bene che gli studenti si impraticiscano con la stesura di codici ricorsivi, scendendo nelle problematiche piu' sottili e complesse, e scoprendo il gusto e la forza ispiratrice dell'eleganza,	Introduzione alla ricorsione – per problemi. Si propongono di effettuare il listing di oggetti per i quali si era prima risolto il problema del conteggio tramite ricorrenze (permutazioni, combinazioni, problema borse delle iOi).	Esercitazione pratica	
	02/02/10	17:00	18:00	1	e' bene che gli studenti si impraticiscano con la stesura di codici ricorsivi, scendendo nelle problematiche piu' sottili e complesse, e scoprendo il gusto e la forza ispiratrice dell'eleganza,	Introduzione alla ricorsione – per problemi. Si propongono i problemi "specchio" ed "alberi" dall'archivio delle iOi.	Esercitazione pratica	

3° giorno	03/02/10	08:00	09:00	1	Sviluppo della capacità di guardare le cose in prospettiva cogliendo la pervasività della ricorsione e riconoscendola come propria forma mentis	Si riconsiderano gli algoritmi di ordinamento visti in precedenza come esempi di approccio ricorsivo. Probabilmente i problemi "specchio" ed "albero" saranno ancora molto caldi, e continueranno a pungolare alcuni dei ragazzi ancora per un po' nella ricerca di soluzioni il più possibile pulite. Ma non appena possibile aprire un nuovo fronte introdurremo Mergesort ad esempio di tecnica divide et impera.	Teoria	
	03/02/10	09:00	10:00	1	Sviluppo dell'efficienza sentita nel corpo e poi anche nella mente. Obiettivi ben più alti si possono raggiungere partendo da questo ambito.	Mergesort – divide et impera. Analisi del tempo di calcolo dell'algoritmo Mergesort, prima euristica e poi dimostrata.	Teoria	
	03/02/10	10:00	11:00	1	Acquisire la tecnica del divide et impera	Problemi sul divide et impera. Va implementato un algoritmo divide et impera (MergeSort). Vogliamo assicurarci che la codifica di algoritmi ricorsivi non presenti particolari incertezze.	Esercitazione pratica	
	03/02/10	11:00	12:00	1	Acquisire la tecnica del divide et impera	Problemi sul divide et impera. Va implementato un algoritmo divide et impera (MergeSort). Vogliamo assicurarci che la codifica di algoritmi ricorsivi non presenti particolari incertezze.	Esercitazione pratica	
	03/02/10	pausa pranzo						
	03/02/10	14:00	15:00	1	Sviluppo della consapevolezza della rilevanza pratica e scientifica delle tecniche apprese e dell'algoritmica. Lo studente deve rendere propria tecnica del divide et impera	Altri esempi di divide et impera. Potrebbero essere esponenziazione veloce, moltiplicazione veloce, e/o anche più semplicemente (ed implementabile!) un approccio più efficiente per il problema del sottointervallo a somma massima.	Teoria	
	03/02/10	15:00	16:00	1	Sviluppo della consapevolezza della rilevanza pratica e scientifica delle tecniche apprese e dell'algoritmica. Lo studente deve rendere propria tecnica del divide et impera	Altri esempi di divide et impera. Potrebbero essere esponenziazione veloce, moltiplicazione veloce, e/o anche più semplicemente (ed implementabile!) un approccio più efficiente per il problema del sottointervallo a somma massima.	Teoria	
	03/02/10	16:00	17:00	1	Introduzione alla programmazione dinamica partendo dal computo dei numeri di Fibonacci.	Introduzione alla programmazione dinamica partendo dal computo dei numeri di Fibonacci.	Teoria	

4° giorno	04/02/10	08:00	09:00	1	<p>Sviluppo della programmazione dinamica della ricorsione speso all'insegna dell'efficienza. E' una tecnica fondamentale che consente di fare cose che si poteva pensare inarrivabili prima di averla mai vista. Invece, vista la su un paio di esempi, lo studente la fa propria. Possedere questa tecnica e' un must per le iOi. Sul piano culturale, essa e' legata a soluzioni classiche come il triangolo di Tartaglia od a tecnologie importanti come l'algoritmo di Viterbi od alle tecniche di editing distance che hanno consentito il sequenziamento del genoma.</p>	<p>Problemi di introduzione alla programmazione dinamica. Potrebbero essere i problemi "Triangolo" e "Palindrome" (delle iOi) o Poldo (della fase regionale). Oppure, a seconda di come si sono dimostrati i ragazzi e della situazione, la prendiamo piu' larga e si procede sull'esempio classico dei numeri di Fibonacci.</p> <p>Intendiamo inoltre percorrere il percorso didattico delineato dal Cormen che arriva alla Programmazione Dinamica attraverso la tecnica della Ricorsione con Memoizzazione lavorando sull'esempio della Massima Sottosequenza Comune. Comunque sia, lo sviluppo dell'argomento richiederà piu' ore. Diverse delle cose viste andranno implementate e sono possibili inoltre interessanti sperimentazioni.</p>	Teoria e Esercitazione pratica
	04/02/10	09:00	10:00	1	<p>Acquisire lo strumento di programmazione dinamica quale tecnica di problem solving. L'esperienza insegna che cio' avviene dando degli esempi su cui possano ragionare.</p>	<p>Problemi di introduzione alla programmazione dinamica. Si procede come delineato sopra.</p>	Teoria e Esercitazione pratica
	04/02/10	10:00	11:00	1	<p>Acquisire lo strumento di programmazione dinamica quale tecnica di problem solving. L'esperienza insegna che cio' avviene dando degli esempi su cui possano ragionare.</p>	<p>Problemi di introduzione alla programmazione dinamica. Si procede come delineato sopra.</p>	Esercitazione pratica
	04/02/10	11:00	12:00	1	<p>Acquisire lo strumento di programmazione dinamica quale tecnica di problem solving. L'esperienza insegna che cio' avviene dando degli esempi su cui possano ragionare.</p>	<p>Problemi di introduzione alla programmazione dinamica. Si procede come delineato sopra. Se ne abbiamo il tempo, introduciamo allora anche alcuni modelli classici come quello dello zaino.</p>	Teoria e Esercitazione pratica
	04/02/10	pausa pranzo					
	04/02/10	14:00	15:00	1	<p>Acquisire lo strumento di programmazione dinamica quale tecnica di problem solving. L'esperienza insegna che cio' avviene dando degli esempi su cui possano ragionare.</p>	<p>programmazione dinamica OR ricorsione/divide et impera. Dobbiamo dare il tempo ai ragazzi di assimilare le tecniche apprese risolvendo autonomamente ulteriori problemi sulla base di esse. Dobbiamo anche dare loro il tempo per implementare nei dettagli alcune soluzioni. Se possibile, procederemo con altri argomenti, ma non posso introdurre la previsione di questo intervallo tampone di un paio di ore per eventualmente riallineare i ragazzi e/o quanto avviene in aula con quanto scritto in questo programma.</p>	Esercitazione pratica

	04/02/10	15:00	16:00	1	Acquisire lo strumento di programmazione dinamica quale tecnica di problem solving. L'esperienza insegna che cio' avviene dando degli esempi su cui possano ragionare.	programmazione dinamica OR ricorsione/divide et impera. Dobbiamo dare il tempo ai ragazzi di assimilare le tecniche apprese risolvendo autonomamente ulteriori problemi sulla base di esse. Dobbiamo anche dare loro il tempo per implementare nei dettagli alcune soluzioni. Se possibile, procederemo con altri argomenti, ma non posso introdurre la previsione di questo intervallo tampone di un paio di ore per eventualmente riallineare i ragazzi e/o quanto avviene in aula con quanto scritto in questo programma.	Esercitazione pratica	
	04/02/10	16:00	17:00	1	Utilizzare il linguaggio dell'ottimizzazione combinatoria dei grafi e digrafi. Fornire gli strumenti teorici e culturali per avvicinarsi a questa disciplina che sta al nocciolo dell'algoritmica e le cui diramazioni inevitabilmente affiorano rilevanti nelle competizioni delle IOI.	Utilizzo dei grafi negli algoritmi di ricerca. Problema del lupo, capra e cavolo. Connessione di un grafo ed algoritmo BFS.	Teoria	
5° giorno	05/02/10	08:00	09:00	1	Accenno al fatto che alcuni problemi si possono risolvere bene ed altri no, con introduzione e riferimento al concetto di buona caratterizzazione e di problemi difficili. Concetti della complessita' computazionale di importanza centrale in algoritmica e fondamentale in informatica.	Cammini Euleriani ed Hamiltoniani.	Teoria	
	05/02/10	09:00	10:00	1	Strutturare i dati fondamentali.	Vari tipi di coda: FIFO, LIFO e priority queue.	Teoria e Esercitazione pratica	
	05/02/10	10:00	11:00	1	Presentare un algoritmo fondamentale per un problema fondamentale. Cosa fa un navigatore?	Computo delle distanze e di cammini minimi con l'algoritmo di Dijkstra.	Teoria	
	05/02/10	11:00	12:00	1	Implementare porta a consolidare. Di fatto solo con l'implementazione siamo sicuri della piena comprensione. Saper mettere in pratica e saper codificare sotto pressione sono abilita' necessarie per le gare. Lo studente riceve feedback su svariati livelli e da piu' soggetti. Con cio' cogliamo inoltre la finalita' metacognitive e di autovalutazione.	Implementazione da parte dei ragazzi dell'algoritmo di Dijkstra.	Esercitazione pratica	
	05/02/10	pausa pranzo						
	05/02/10	14:00	15:00	1	Affrontare il tema del problema classico, di una struttura fondamentale quale un albero di connessione. Una tecnica fondamentale di progetto di algoritmi che e' importante imparare ad usare e a non impiegare a sproposito, e poi di nuovo ad usarla convenientemente.	Computo di alberi di peso minimo. Approccio greedy	Teoria	

	05/02/10	15:00	16:00	1	Affrontare il tema del problema classico, di una struttura fondamentale quale un albero di connessione. Una tecnica fondamentale di progetto di algoritmi che e' importante imparare ad usare e a non impiegare a sproposito, e poi di nuovo ad usarla convenientemente.	Computo di alberi di peso minimo. Approccio greedy	Teoria
	05/02/10	16:00	17:00	1	Grafi come linguaggio per l'ottimizzazione combinatorica, (e per l'informatica, e per la ricerca operativa)	grafi bipartiti, buona caratterizzazione ed algoritmo di riconoscimento. Oppure problemi combinatorici su grafi.	Esercitazione pratica
	05/02/10	17:00	18:00	1	Approfondire i concetti di buona caratterizzazione di problemi difficili. Concetti della complessita' computazionale di importanza fondamentale ed impatto rivoluzionario nella matematica moderna ma anche sul piano filosofico. E fonte metodologica e guida per il problem solving	grafi bipartiti, buona caratterizzazione ed algoritmo di riconoscimento. Oppure problemi combinatorici su grafi.	Esercitazione pratica
6° giorno	06/02/10	09:00	10:00	1	Sviluppare della capacita' di autodiagnosi	valutazione dell'esperienza	Esercitazione pratica
	06/02/10	10:00	11:00	1	Sviluppare della capacita' di autodiagnosi	Test finale	Teoria
	06/02/10	11:00	12:00	1	Sviluppare della capacita' di autodiagnosi	valutazione dell'esperienza	Teoria
Totale ore:				40			